Development of Some Scalable Pattern Recognition Algorithms for Real Life Data Analysis

Partha Garai

Machine Intelligence Unit
Indian Statistical Institute
Kolkata - 700 108, India

A thesis submitted to Indian Statistical Institute
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Computer Science

2017

Dedicated to my Parents
Basudeb Garai and Manasi Garai

Anything good that has come to my life
has been because of your
guidance, encouragement, and love.

Acknowledgment

At the very outset, I express my sincere gratitude to my supervisor, Dr. Pradipta Maji, for his invaluable advice, support and friendship on both an academic and a personal level. His attention, moral support and timely suggestions were extremely valuable for me in the completion of my research work. I am also thankful to my supervisor for directing me towards this research topic. He is Dr. Maji, who made me understand that hard work pays off. He has also taught me to be disciplined and loyal in research and in life.

I would like to record my sincere gratitude and appreciation to the Machine Intelligence Unit, Indian Statistical Institute, Kolkata for allowing me to work in a great environment during this period. I am very greatful to Prof. S. K. Pal, Prof. M. K. Kundu, Prof. C. A. Murthy, Prof. A. Ghosh, and Dr. K. Ghosh of the Machine Intelligence Unit, for their helpful suggestions and comments during my progress report presentations. My very special thanks go to all the staff members of Machine Intelligence Unit, for making all possible facilities available to me.

I am really grateful to Prof. Rajat K. De, Head, Machine Intelligence Unit. I would like to thank all my colleagues in Machine Intelligence Unit and Biomedical Imaging and Bioinformatics Lab (BIBL) for continuously sharing my load so that I could do my thesis work smoothly. I wish to thank all of my colleagues of Kalyani Government Engineering College, for supporting me in my work. Beside the academics, a brilliant research environment has also been created by them, which helped me a lot. I would like to specially thank my dearest friend Chaitali and Santi da for providing me unconditional support. They have been wonderful. I am also thankful to Department of Science and Technology (DST), Government of India, for providing me a partial financial assistance to carry out this research work.

I wish to thank God for putting hurdles in my life that has helped me realizing my potential. The love and blessings of my father, who passed away untimely, is greatly acknowledged. I would like to express my deepest gratitude to my mother. It is her encouragement and blessings that enabled me to cross these hurdles. A special note of thanks is also due to my wife and all the relatives who are anxiously awaiting for the thesis to be published. Much of what I have learned over the years has come from my friends and well wishers, both in academic and non-academic spheres. To all such people I convey my sincere thanks.

.....

Partha Garai

June, 2017

Machine Intelligence Unit,

Indian Statistical Institute, Kolkata, India.

Abstract

A huge amount of data is being generated continuously as a result of recent advancement and wide use of high-throughput technologies. With the rapid increase in size of data distributed worldwide, understanding the data has become critical. In this regard, dimensionality reduction and clustering have become the necessary preprocessing steps of multiple research areas and applications. One of the important problems of real life large data sets is uncertainty. Some of the sources of this uncertainty include imprecision in computation and vagueness in class definitions. The uncertainty may also be present in the definition of class membership function.

In this background, the thesis addresses the problem of dimensionality reduction and clustering of real life data sets, in the presence of noise and uncertainty. The thesis first presents the problem of feature selection using both type-1 and interval type-2 fuzzy-rough sets, which are effective for dimensionality reduction of real life data sets when uncertainty is present in the data set. The properties of fuzzy-rough sets allow greater flexibility in handling noisy and real valued data. While the concept of lower approximation and boundary region of rough sets deals with uncertainty, incompleteness, and vagueness in class definition, the use of either type-1 or interval type-2 fuzzy sets enables efficient handling of overlapping classes in uncertain environment. Moreover, a new concept of *"simultaneous attribute selection and feature extraction"* is introduced for dimensionality reduction, integrating judiciously the merits of both feature selection and extraction.

A scalable rough-fuzzy clustering algorithm is introduced for large real life data sets, where the theory of rough hypercuboid approach, interval type-2 fuzzy sets, and c-means algorithm are integrated judiciously to handle the uncertainty present in a data set. While the concept of rough hypercuboid approach deals with uncertainty, incompleteness, and vagueness in cluster definition, the use of fuzzy membership of interval type-2 fuzzy sets in the boundary region of a cluster enables efficient handling of overlapping partitions in uncertain environment. Finally, the application of both clustering and feature selection algorithms is demonstrated by grouping functionally similar microRNAs from microarray data. The proposed approach can automatically select the optimum set of features while clustering the microRNAs, making the complexity of the algorithm lower.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Definition

The modern era is often emphasized with the term "digital world", where massive amount of data are generated and collected regularly in every aspects of human life, such as, online transactions, performance analysis, banking, communications, and so on. Government agencies, commercial, scientific, and business organizations are regularly using computers, not only for computation purposes, but also to store the large volume of data that they generate over time, or gather from different sources. The data may come in a wide variety of forms, numeric, textual, pictorial, symbolic, and aural, which may be full of errors, redundancy, imprecision, and so on. With the help of high speed and well connected computer networking, such data is now being easily accessible to increased number of people. Truly speaking, we are in the midst of a "data deluge" and some automated procedure is required to understand and process this vast and diverse collection of data, possibly in real-time mode. This issue has earned a new phrase, called data mining, which is aimed at finding out some natural structures present within the immense volume of data, which may be of heterogeneous type. The main goal of data mining is the knowledge discovery by generalizing and applying some existing and new pattern recognition algorithms, which involve the methods of artificial intelligence, machine learning, statistics, and database systems. Hence, pattern recognition plays a vital role in data mining. Data mining deals with the process of extracting information from a data set and transforming it into a valid, novel, potentially useful, and understandable structure for further use. Hence, it can be considered as an application of pattern recognition and machine learning principles in the context of voluminous data sets containing possibly heterogeneous data.

Dimensionality reduction [101, 112] is one of the important problems encountered in pattern recognition, machine learning, and data mining. It refers to a process of selecting a map by which a sample in an m-dimensional measurement space is transformed into an

object in a d-dimensional feature space, where $d < m$. In general, the high dimensional real life data sets contain a large number of irrelevant and redundant or insignificant attributes, which increases processing as well as storage cost while decreasing the overall performance. So, the main objective of dimensionality reduction is to retain the optimum salient characteristics necessary for the pattern recognition process and to reduce the dimensionality of the measurement space so that effective and easily computable algorithms can be devised for efficient classification. The problem of dimensionality reduction has two aspects, namely, formulation of a suitable criterion to evaluate the goodness of a feature set and searching the optimal feature set in terms of the criterion. The algorithm used for dimensionality reduction can be broadly classified into two categories, namely, feature selection in the measurement space and feature selection in a transformed space. The techniques in the first category generally reduce the dimensionality of the measurement space by discarding redundant or least information carrying features [61]. On the other hand, those in the second category utilize all the information contained in the measurement space to obtain a new transformed space, thereby mapping a higher dimensional pattern to a lower dimensional one. This is referred to as feature extraction [101, 112]. Feature extraction techniques usually provide a richer feature subset than a feature selection algorithm, but with a higher cost. Dimensionality reduction projects a higher dimensional data set onto smaller dimensions. So, it is useful for data visualization also. The technique is also important to overcome the overfit condition in the training phase. When number of training samples is limited, huge number of features can overfit the training data and loose the generalization capability of the sample set.

On the other hand, clustering tries to find the natural groups present in a data set, without knowing or explaining why they exist. Here, a set of objects is partitioned into several groups so that the objects in the same group or cluster are more similar to each other than to those in other clusters [216, 262]. Popular notions of clusters include groups with low distances among the cluster members, dense areas of the data space, intervals or particular statistical distributions. The appropriate clustering algorithm and the parameters required depend on the data set and the exact use of the result. The outcome of clustering depends on the criterion used, which is related to the final aim of the clustering. Consequently, the criterion must be supplied by the user, so that the result of the clustering matches with his/her need. In case of large number of objects of similar types, the cluster means or medians may be the representative of the clusters, which is an useful technique to reduce the size of the sample space.

One of the major problems in real life data analysis is the management of uncertainty, which arises from incompleteness and vagueness in class definitions. In this background, the possibility concept introduced by the fuzzy sets theory and rough sets theory has gained popularity in modeling and propagating uncertainty. While the membership of fuzzy sets

enables efficient handling of overlapping partitions, the rough sets deal with uncertainty, vagueness, and incompleteness in class definition. They provide mathematical frameworks to capture uncertainties associated with the data. They are also complementary in some aspects. The objective of the current research work is to investigate, both theoretically and experimentally, the relevance of rough-fuzzy approaches to pattern recognition with real life applications. Various methodologies are developed for feature selection and clustering, where emphasis is given on handling data sets, which are large in terms of both sample size and dimension and involve classes that are overlapping, interactable, and/or having nonlinear boundaries, and demonstrating their success in real life applications.

1.2 Limitations of Current Methods

The theory of rough sets can be used to find a subset of informative features from the original attributes of a given data set with discretized attribute values [83, 327]. However, the data sets generated from the real world applications usually contain real valued data and fuzzy information. In rough set theory, the real valued features are divided into several discrete partitions and the dependency or quality of approximation of a feature is calculated. The inherent error that exists in discretization process is of major concern in the computation of the dependency of real valued features. To prevent the loss due to this discretization process, fuzzy and rough sets can be combined to construct fuzzy-rough sets, which provide an important direction in reasoning with uncertainty for real valued data sets, and can easily be used for feature selection. However, in case of large data sets, the complexity of the currently available feature selection methods using fuzzy-rough sets increases noticeably, which prevents the use of the current methods in such real life applications. So, some new methods need to be proposed to deal with the issues mentioned above, which can efficiently decrease the number of features with a relatively low computational complexity.

The existing fuzzy and fuzzy-rough set based feature selection algorithms, available in the literature, use type-1 (T1) fuzzy sets [556]. Although the researchers found some limitations of T1 fuzzy sets, most of the contributions in this domain are still restricted to this theory. Mendel [342] pointed out at least four sources of uncertainties present in T1 fuzzy logic systems. All the uncertainties [556] converge into uncertainties about the membership functions of fuzzy sets, which cannot be handled by T1 fuzzy logic. This is also one of the major limitations of the current fuzzy-rough set based dimensionality reduction methods. So, some new techniques are required to address this problem.

Apart from the feature selection, which selects some features from the original measurement space, another useful technique of dimensionality reduction is feature extraction, which extracts features by transforming the measurement space. In general, a feature ex-

traction technique [257, 483] provides a richer feature subset than that produced by a feature selection algorithm, but with an increased cost. Hence, for a given data set, it is not very easy to decide whether to select a feature from original measurement space or extract a new feature from the transformed space. So, some new methods need to be developed that can simultaneously select original features of measurement space and new extracted features of transformed space.

On the other hand, the rough-fuzzy clustering is one of the important techniques in clustering, where each cluster is represented by a core region and a boundary region. The rough-fuzzy clustering is effective for the data set where clusters are vaguely defined and boundaries are overlapping in nature. One of the main problems of rough-fuzzy clustering is how to decide the core and boundary regions of a cluster. In general, the objects are put into core or boundary regions depending on some user defined threshold values. This decision-making step plays a vital role in the performance of the algorithm. The requirement of user specified values is another major source of concern of the existing rough-fuzzy clustering. Moreover, the traditional rough-fuzzy clustering works well when the class membership functions are known. But, when uncertainty is present to define the fuzzy membership functions, conventional rough-fuzzy clustering algorithms are not useful in grouping the samples properly.

1.3 Scope and Organization of the Thesis

The major focus of this research work is to devise some new feature selection and clustering methodologies, which can deal with some of the limitations mentioned above and are efficient in terms of computational complexity and accuracy. The outline of the thesis is presented in Figure 1.1.

Chapter 1 provides an introduction to feature selection and clustering in the domain of pattern recognition. The limitations of currently available methods, along with the importance of both rough sets and fuzzy-rough sets for feature selection and clustering, are also described in this chapter. Chapter 2 presents the basic concepts of pattern recognition and their applications in day-to-day life. Different statistical and soft computing approaches for pattern recognition, along with some existing fuzzy-rough set based approaches for pattern recognition, are also highlighted in this chapter.

Prior to analysis of the data set, selecting or extracting relevant and significant features is an important preprocessing step for pattern recognition. In this regard, Chapter 3 presents a feature selection method based on fuzzy-rough sets. It selects features by maximizing both relevance and significance of the selected features. Different feature evaluation criteria such as dependency, relevance, redundancy, and significance for attribute selection task using fuzzy-rough sets are also presented. The performance of different rough set

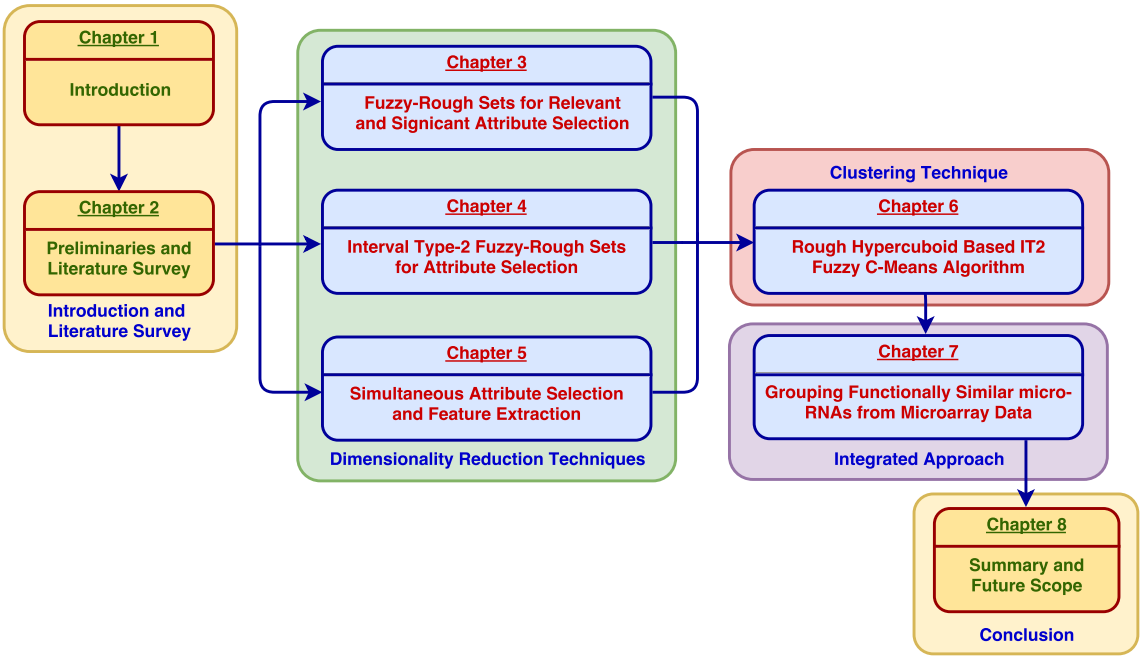


Figure 1.1: Outline of the thesis

models is compared with that of some existing feature evaluation indices based on the predictive accuracy of nearest neighbor rule, support vector machine, and decision tree. The effectiveness of fuzzy-rough set based attribute selection method, along with a comparison with existing feature evaluation indices and different rough set models, is demonstrated on a set of benchmark and microarray gene expression data sets.

Although fuzzy-rough set is a well-known technique for modeling the uncertainties in the feature space, it is not effective in a situation where determination of exact membership function for any fuzzy set is difficult. In this regard, an interval type-2 (IT2) fuzzy-rough set based feature selection method is presented in Chapter 4, where the features are selected by maximizing both relevance and significance of the features. By introducing the concept of lower and upper fuzzy equivalence partition matrices, the lower and upper relevance and significance of the features are defined for IT2 fuzzy approximation spaces. Different feature evaluation criteria such as dependency, relevance, and significance are presented for attribute selection task using IT2 fuzzy-rough sets. The performance of IT2 fuzzy-rough sets is compared with that of some existing feature evaluation indices including classical rough sets, neighborhood rough sets, and type-1 fuzzy-rough sets. The effectiveness of the proposed IT2 fuzzy-rough set based attribute selection method, along with a comparison with existing feature selection and extraction methods, is demonstrated on several real life data sets.

In general, a feature extraction technique provides a richer feature subset than that obtained using a feature selection algorithm with a higher cost. Hence, for a given data

set, it is very difficult to decide whether to select a feature from the original measurement space or extract a new feature by transforming the existing features. In this regard, a novel dimensionality reduction method, based on fuzzy-rough sets, is presented in Chapter 5 that simultaneously selects attributes and extracts features using the concept of feature significance. The proposed method is based on maximizing both relevance and significance of the reduced feature set, whereby redundancy therein is removed. The proposed dimensionality reduction method, based on fuzzy-rough sets, is shown to be more effective for generating a relevant and significant feature subset. The effectiveness of the proposed dimensionality reduction method, along with a comparison with existing attribute selection and feature extraction methods, is demonstrated on several real life data sets.

The rough-fuzzy clustering is one of the most promising techniques for grouping data, where the concept of lower and upper approximation of rough sets is coupled with fuzzy set theory. One of the major issues of rough-fuzzy clustering is how to decide whether some objects are to be put in lower approximation or boundary region of a cluster. Moreover, conventional rough-fuzzy clustering technique is inefficient when the determination of fuzzy membership function is difficult. In this regard, a novel clustering algorithm is formulated in Chapter 6. It integrates judiciously the theory of rough hypercuboid approach, interval type-2 fuzzy sets, and c-means algorithm, to approximately manage the uncertainty present in a data set. Using the concept of hypercuboid equivalence partition matrix of rough hypercuboid approach, the lower approximation and boundary region of each cluster are implicitly found out, without the need of any user specified threshold. The interval valued fuzzifier is used to deal with the uncertainty associated with the parameters of rough-fuzzy clustering algorithms, where determining the appropriate value of the fuzzifier is difficult. The problems of initial prototype selection and stuck in local minima of different c-means algorithms are efficiently solved by developing a robust prototype selection method. The effectiveness of the proposed interval type-2 rough-fuzzy clustering algorithm, along with a comparison with other related algorithms, is demonstrated both qualitatively and quantitatively on real life data sets.

Chapter 7 presents an application of clustering in the domain of bioinformatics. It refers to grouping functionally similar microRNAs (miRNAs) from miRNA data sets. Each miRNA data set contains large number of samples, where each miRNA consists of large number of time points or features. While there exist many algorithms for clustering, the important issue of feature selection, that is, which attributes of the data should be used by the clustering algorithms, is rarely touched upon. Feature selection for clustering is difficult as there is no class label for the data and, thus, no obvious criterion to guide the search. However, the performance of both clustering and feature selection is dependent on the incorporated clustering algorithms, which may be sensitive to their initializations and suffer from locally optimal solutions. In this regard, in Chapter 7, the clustering algo-

rithm proposed in Chapter 6 is judiciously combined with the feature selection algorithm proposed in Chapter 4 to develop an integrated method, which can select the optimum set of features while clustering the samples, keeping the complexity of the algorithm in mind. Comparisons are performed with other existing algorithms to prove the effectiveness of the proposed algorithm for grouping miRNA data sets.

Finally, Chapter 8 concludes the thesis and discusses the future scopes of improvements of the proposed research work. Through the aforementioned investigations and experiments, the potential utility of fuzzy-rough sets for dimensionality reduction and clustering is demonstrated by developing of some scalable pattern recognition algorithms for real life data analysis.

Chapter 2

Preliminaries and Literature Survey

This dissertation reports several new contributions in the field of pattern recognition for real life large data sets. In this background, the first part of this chapter covers the basic concepts of pattern recognition and their applications in diverse fields. The following sections cover the statistical and soft computing based approaches. The last section briefly describes different approaches of dimensionality reduction and clustering.

2.1 Introduction to Pattern Recognition

The task of pattern recognition [37, 102, 237, 464] has been encountered in a wide range of human activities. We are performing the task of pattern recognition at every instance of our working lives. We recognize the objects around us, and move and act in relation to them. Typical examples include recognizing the voice of a friend over the phone, or the flavor of an ice cream, reading news, driving a car, diagnosing a disease, and distinguishing a piece of music played on a sitar from that on a sarod, etc. In a broader perspective, the term actually cover any context in which some decision or forecast is made on the basis of currently available information. The problem deals with the construction of a procedure to be applied to a set of inputs; the procedure assigns a new input to one of a set of classes on the basis of observed attributes or features. The construction of such a procedure on an input data set is defined as pattern recognition. So, pattern recognition is a subject of researching object description and classification methods; it is also a collection of mathematical, statistical, heuristic, and inductive techniques of fundamental role in executing the tasks like human being on computers. In a sense, pattern recognition is figuring out actual problems via mathematical methods.

2.1.1 Methods: Dimensionality Reduction, Clustering and Classification

Machine recognition [112, 498] of patterns can be viewed as a twofold task, consisting of learning the invariant and common properties of a set of samples characterizing a class for deciding a new sample as a possible member of the class by noting that it has properties common to those of the set of samples. In other words, pattern recognition can be described as a transformation from the measurement space M to the feature space F and finally to the decision space D, that is, $M \rightarrow F \rightarrow D$. Here, the mapping $\delta : F \rightarrow D$ is the decision function and the elements $d \in D$ are termed as decisions. In a general setting, the process of pattern recognition is visualized as a sequence of few steps:

1. Data acquisition [175],
2. Preprocessing [175, 493],
3. Dimensionality reduction [43, 68, 231, 254],
4. Clustering [112, 216],
5. Classification [175, 493].

At the first step, depending on the environment within which the objects are to be classified, data are gathered via a series of sensors. After the data acquisition phase, few preprocessing tasks, such as, noise reduction, filtering, encoding, and enhancement are applied on the collected data for extracting pattern vectors. Afterwards, a feature space is constituted in order to reduce the space dimensionality. However, in a broader perspective, this stage significantly influences the entire recognition process. Finally the classifier is constructed, or in other words, a transformation relation is established between features and classes. The tasks of data acquisition, feature selection, clustering, and classification are discussed below. Details of different aspects of pattern recognition, namely, data acquisition, preprocessing, feature selection/extraction, classifier design, cluster analysis, learning and their various applications are available in the literatures [48, 51, 112, 238, 391, 498].

2.1.1.1 Data Acquisition

In this phase, data are gathered [175, 493] from the environment via a set of sensors. Pattern recognition techniques can be used in various domains, correspondingly the data can be quantitative, qualitative, or both; they also may be linguistic, numerical, pictorial, or any combination thereof. Usually, two types of data structures are used for pattern recognition systems: object data vectors and relational data. In the first type, sets of numerical vectors having m features are represented as $x = \{x_1, x_2, \cdots, x_n\}$, a set of n

objects in the m-dimensional measurement space $\mathbb{R}^m$. The ith object is represented as x_i, $i = 1, 2, \cdots, n$; x_{ij} is the jth ($j = 1, 2, \cdots, m$) feature corresponding to the object i. Relational data are represented with a set of n^2 numerical relationships between pairs of objects i and k, say, r_{ik}. In other words, r_{ik} represents the extent to which objects i and k are related like some binary relationship ρ. If the objects in $O = \{o_1, o_2, \cdots, o_n\}$ are pairwise related by ρ, then $\rho : O \times O \rightarrow \mathbb{R}$.

2.1.1.2 Preprocessing

After collecting the data, multiple data preprocessing techniques [175] are applied on the acquired data to extract the corresponding pattern vectors. As data are originated from multiple and heterogeneous sources of typically huge sizes, they may contain noisy, missing, and inconsistent measurements. To get useful data mining results, the quality of data must be improved. Data preprocessing is the next step, which is divided into the following parts: data cleaning, data integration, and transformation. Data cleaning can be applied for removing noise and correcting inconsistencies in the data, where as, data integration merges data coming from different sources into a single data set. The data transformation process transforms the data into several different forms that is suitable for mining. For example, normalizing the data may improve the accuracy and efficiency of mining algorithms containing length and distance values.

2.1.1.3 Dimensionality Reduction

Dimensionality reduction [34, 101] is another important step in any automatic pattern recognition process. The main objective of dimensionality reduction is to retain the optimum salient characteristics necessary for the recognition process and to reduce the dimensionality of the measurement space, so that effective and easily computable algorithms can be devised for efficient classification. The problem of dimensionality reduction has two aspects: the formulation of a suitable criterion to evaluate the goodness of a feature and the selection of optimal subset of the available features. The major mathematical measures so far devised for the estimation of feature quality are mostly statistical in nature, and can broadly be classified data into two categories: (i) feature selection in the measurement space [57,169,231,254]; and (ii) feature extraction in the transformed space [43,68,103,209]. The techniques in the first category generally reduce the dimensionality of the feature set by discarding redundant information carrying features. On the other hand, those in the second category utilize all the information contained in the pattern vectors, and map a higher-dimensional pattern vector to a lower-dimensional one.

2.1.1.4 Clustering

Clustering [45, 96, 112, 132, 216, 447, 572] or cluster analysis finds clusters or groups of data objects that are similar to one another in some sense. The members of a cluster are more similar to each other than that of the members of other clusters. The goal of clustering analysis is to find high-quality partitions so that the inter-cluster similarity is low and the intra-cluster similarity is high. Clustering is useful for exploring data. If there are many cases and no obvious grouping, clustering techniques can be used to find natural groupings. Clustering can also serve as an useful data-preprocessing step to identify homogeneous groups on which the supervised models can be built. Clustering is also used to segment [49, 151, 177, 227, 388] the data. Clustering algorithms model segmented data into groups that were not previously defined. Clustering can also be used for anomaly detection. Once the data has been segmented into clusters, same cases may be found out which do not fit well into any clusters. These cases are treated as anomalies or outliers.

2.1.1.5 Classification

The problem of classification [44,81,112,158,374] is basically to partition the feature space into regions, one region for each category of input. Thus, it attempts to assign every data point in the entire feature space to one of the possible classes. Data classification is a two step-process. In the first step, a classifier is built, describing a predetermined set of data classes or concepts. This is the learning step or training phase, where a classification algorithm builds the classifier by analyzing or learning from a training set. In the next step, the model just built is used for classification. Here, the predictive accuracy of the classifier is measured. If the training set is used to compute the accuracy of the classifier, the estimate would likely to be optimistic, as the classifier tends to overfit the data (that is, it may also learn some specific anomalies of the training data during training, which are absent in the complete data set). Therefore, a completely different set is used for testing, called as test set, made up of test samples and their corresponding class labels. These samples are randomly selected from the general data set and are not related with the training tuples, that means, the tuples are not used for training the classifier. The accuracy of a given classifier on a test set is the percentage of test samples which are correctly classified.

2.1.2 Applications of Pattern Recognition

There are many important fields of applications where different pattern recognition techniques can be applied.

2.1.2.1 Applications of Dimensionality Reduction

The techniques of feature selection have several interesting applications in diverse fields. Few of them are mentioned below.

Text Mining Text mining is a section of data mining, where a document is represented by the model, called bag-of-words. In this model, each document actually corresponds to the counts of words occurring in that document. Usually, the feature vectors are formed, such that, each feature (each element of the feature vector) holds the count of a specific word. Another alternative is just to indicate the presence/absence of a word using boolean value. Vocabulary represents the set of words, whose occurrences are stored. As the volume of documents increases with different kinds of words, the size of the feature vector increases, increasing the complexity and processing time to search any desired words or a sequence from the whole data set. To build the vocabulary, the words may be collected first from all the documents in the data set and then feature selection can be used to prune the vocabulary.

- **Text Classification:** The goal of text categorization [8, 131] is the classification of documents [302, 304, 449, 481, 506] into a fixed number of predefined categories. Each document can be in multiple, exactly one, or no category at all. Using machine learning, the objective is to learn classifiers from examples which perform the category assignments automatically. To increase the speed of processing, a set of best features need to be found out that can help the categorization process.

- **Text Clustering:** Document clustering (or text clustering) [54, 296, 482, 497] is the application of cluster analysis to textual documents [185, 307, 543, 550]. It has applications in automatic document organization, topic extraction and fast information retrieval or filtering. Researches proved that the feature selection can improve the performance and efficiency of text clustering by finding out the features that are perfect to compute the similarity/dissimilarity between the documents.

Image Processing and Computer Vision Representing images is not a straightforward task, as the number of possible image features is practically unlimited [55]. The choice of features typically depends on the target application.

- **Image Classification:** Image classification [17,55,277,552] includes a broad range of decision-theoretic approaches to the identification of images (or parts thereof) [120, 258, 309, 559]. In all classification algorithms, it is assumed that the given image consists of one or more features and each of these features belongs to one of several distinct and exclusive classes. The classes may be specified a priori by

the domain expert (as in supervised classification) or can be found automatically (as in unsupervised classification), where the expert may only specify the number of desired categories. The best feature subset needs to be found out which produces the best classification result.

- **Robotic Vision:** Robotic vision [375, 448] includes different methods for processing, analyzing, and understanding. For a robot, all these methods produce some information which can be used to take the decisions. The input information are translated into features, where the best features are required to identify the objects or to choose a specific path for movement.

Bioinformatics An interesting application of feature selection is in biomarker discovery from genomics data. In genomics data, individual features correspond to genes, so by selecting the most relevant features, one gains important knowledge about the genes that are the most discriminative for a particular problem.

- **Biomarker Discovery:** A biomarker [9,100,182,259,278,452,488] is a measurable indicator of the severity or presence of any disease state. More generally, a biomarker is anything that can be used as an indicator of a particular disease state or some other physiological state of an organism. Suitable biomarker can be selected by comparing them, taking into account the predictive performance and stability of the selected gene sets.

- **Microarray Gene Expression Data Classification:** DNA microarray [181,203, 229,367,467,504,507,532,562] technology provides tools for monitoring the expression levels of huge number of different genes simultaneously. One salient application of gene expression microarray data [3] is the classification of biological samples or prophecy of clinical and other outcomes, based on the features present among the genes. As all the features are not equally important, the most appropriate genes must be selected using feature selection.

Industrial Applications and Automation

- **Fault Diagnosis:** Feature selection is important in fault diagnosis [35, 73, 78, 174, 211, 290, 308, 402] in industrial applications, where numerous redundant sensors monitor the performance of a machine. The data generated contain redundant and misleading features. The accuracy of detecting a fault (i.e. solving a binary classification problem of machine state as faulty vs. normal) can be improved by using feature selection.

- **Brain-Computer Interface:** Brain-computer interfaces [153, 425, 428, 561] allow human being to manipulate computers and machinery with their thoughts. Various researches are going on and several commercial devices are available that can transmit signals directly to/from someone's brain allowing him/her to see, hear or feel specific sensory inputs and instruct accordingly. Appropriate features are required to sense and interface with the devices.

2.1.2.2 Applications of Clustering

Clustering has several applications in multiple domains. Few of them are specified below.

Computer Science In the domain of computer science, clustering techniques have several applications. Few of them are mentioned below.

- **Software Evolution:** In case of software evolution [31, 121, 178, 263, 284, 421, 432, 440], clustering is an useful technique, which reforms the functionality to reduce legacy properties in code, that has become dispersed. It can be considered as restructuring and hence, is a way of direct preventative maintenance.
- **Image Segmentation:** A digital image can be partitioned into distinct regions [130,281,285,435] with the help of clustering, which can be used for border detection or object recognition [22, 143, 157, 369, 466].
- **Evolutionary Algorithms:** Clustering technique can help to identify different niches within the population of an evolutionary algorithm [97, 144, 156, 179, 300, 456, 473, 476, 490] so that reproductive opportunity can be distributed more evenly amongst the evolving species or subspecies.
- **Anomaly Detection:** Anomalies/outliers [77, 249, 251, 379, 463, 471, 540] can be identified explicitly or implicitly with the help of clustering algorithms.

Social Science Clustering techniques are also equally important in the field of social science. Some specific cases are explained.

- **Crime Analysis:** To identify areas having greater incidences of particular types of crime [62,465], clustering techniques may be used. Law enforcement can be easily and efficiently managed if these distinct areas or "hot spots" are identified, where a similar crime has happened over a period of time.
- **Educational Data Mining:** Cluster techniques can be used to identify the schools or students having similar properties [26, 444].

- **Typologies:** From poll data [245], cluster analysis can be used to discern typologies of opinions, habits, and demographics that may be useful in politics and marketing.

World Wide Web Web analysis and web mining techniques are also equally important.

- **Social Network Analysis:** Clustering techniques can be used in the study of social networks [12,125,298,404,472], to identify communities and connections among large groups of people.
- **Search Result Grouping:** In case of grouping of the files and websites [200, 276, 280] intelligently, clustering techniques can be used to form a more specific and relevant set of search results compared to general purpose search engines like Google. Different web based clustering tools are currently available, such as Clusty.
- **Slippy Map Optimization:** Clustering techniques are used in Flickr's map of photos and other web sites to reduce the number of markers on a map [363]. This technique makes the pages to load faster, while reducing the amount of visual clutter.

Biology, Computational Biology and Bioinformatics Clustering techniques are now getting importance in these fields. Few specific cases are listed below.

- **Plant and Animal Ecology:** Clustering techniques are used to generate and analyze the spatial and temporal distribution of communities (assemblages) of organisms [434] in heterogeneous environments. It is also helpful for plant systematics to generate artificial phylogenies or clusters of organisms (individuals) at the species, genus or higher level that share a number of attributes.
- **Transcriptomics:** Clustering is used to group the coexpressed genes [180]. The groups usually contain functionally related proteins, such as enzymes, for a specific pathway [236,484], or genes that are co-regulated. High throughput experimentations are used for expressed sequence tags (ESTs) [399] or DNA microarrays, which are very powerful tools for genome annotation.
- **Sequence Analysis:** Homologous sequences [140, 282, 377] can be grouped into gene families using clustering techniques, which is now a very important concept in bioinformatics and evolutionary biology.
- **Human Genetic Clustering:** Clustering analysis can be used to find the similarity of genetic data [474] for inferring population structures.

Medicine and Health Care Clustering has vast application in medicine and health care. Some specific examples are given.

- **Medical Imaging:** Cluster analysis can be used on medical images such as PET, MRI, X–Ray scans [24, 264] to identify different types of tissues from a three-dimensional image [40, 164].
- **Drug design and Testing:** Drug design [162] is an iterative process requiring cycles of compound synthesis and testing, with each successive synthesis phase yielding molecules predicted to have improved characteristics over the previous set of compounds.

Business and Marketing Business and marketing is a new and wide application platform of clustering, where the efficient and timely prediction is extremely valuable.

- **Market Research:** Cluster analysis is widely used in market research [46, 244, 509, 512] for partitioning the general buyers into different market segments to better understand the relationships between different groups of consumers, and for use in market segmentation, item positioning, new product development and selecting potential buyers.
- **Grouping of Shopping Items:** All the shopping items [549, 573] available on the web can be grouped into a set of unique products using clustering techniques. For example, all the items on Amazon are grouped into some product categories so that the buyers can find them easily.

2.1.2.3 Applications of Classification

Classification techniques are having applications in various areas of our day-to-day life. Few of them are mentioned below.

Computer Vision Classification techniques are really important in the domain of computer vision, which is a domain to recognise various objects automatically, just like human being. Few specific applications are mentioned below.

- **Medical Imaging and Analysis:** Using the medical imaging technique [18, 135, 291, 350, 376, 530, 568], the interior of a body can be represented visually for clinical analysis and medical intervention. It is also used to visualize the functions of some organs or tissues. The different parts of image need to be automatically classified into some predefined classes.

- **Optical Character Recognition:** The images of typed, handwritten or printed text are segmented and classified into machine-encoded text [90,362,378,544,558]. It works on different types of images, like, a scanned document, a photo of a document (photo of number plate of a car), a scene-photo or from subtitle text superimposed on an image.
- **Video Tracking:** Video tracking [23, 118, 360, 469, 514, 566] is now being used in multiple domains, some of which are human-computer interaction, security and surveillance, video communication and compression, augmented reality, traffic control, medical imaging and video editing, where the "tracking" is actually finding and classifying the moving objects to some predefined classes.

World Wide Web The classification techniques are widely used in internet now-a-days. The unknown objects are required to be classified into a particular type so that it can be stored or retrieved quickly.

- **Document Classification:** In document classification [5, 53, 243, 420, 492, 535] or document categorization problem, the task is to assign a document to one or more classes or categories. Traditionally, it was done manually, but, due to the arrival of tremendous amount of documents which are available in web now, the computer based automatic classification becomes the only way to address the issue.
- **Internet Search Engines:** A web search engine [338, 510, 570, 575] is used to search information in the World Wide Web. The information may be a mix of web pages, images, and other types of files. The task is to assign a web search query to one or more predefined categories, based on its topics.

Drug Discovery and Development In the fields of medicine, drug discovery is extremely important, by which, new candidate medications are discovered.

- **Toxicogenomics:** Toxicogenomics [159, 166, 295, 503, 579] is a domain which is related with the collection, interpretation, and storage of information about gene and protein activity within particular cell or tissue of an organism in response to toxic substances. The analysis involves a wide array of bioinformatics and statistics, and classification approaches.
- **Quantitative Structure-Activity Relationship:** Quantitative structure-activity relationship (QSAR) models [163, 294, 417, 426] are regression or classification models used in the chemical and biological sciences and engineering. The classification QSAR model relates the predictor variables to a categorical value of the response variable.

Security and Authentication

- **Biometric Identification:** Biometric authentication and identification [15, 19, 87, 160] offers a wide range of possible solutions, which could be used to authenticate users and thus to provide an extra level of security and theft prevention.
- **Speech Recognition:** Speech recognition [126, 337, 480, 485, 551] is related to the recognition and translation of spoken language into text by computers. Usually, training phase is the first step, where a speaker reads text or isolated vocabulary into the system. The system analyzes the person's specific voice and uses it to fine-tune the recognition of that person's speech, resulting in increased accuracy. Classification technique is used to classify the person's spoken word to a textual word.

2.1.3 Different Pattern Recognition Approaches

There are several well-known approaches for pattern recognition. Each of them is briefly described next.

2.1.3.1 Statistical Pattern Recognition

Statistical approach [141] is widely used, where the patterns are represented in terms of m features or measurements and considered as a point in an m-dimensional space. The objective is to select those features which makes the objects of different categories to occupy compact and disjoint regions in the feature space. The effectiveness of the technique is measured by how well the objects from different classes can be separated. For a given set of training patterns, the objective is to find out decision boundaries with the help of the feature space, that can separate the objects to different classes. In case of statistical decision theoretic approach, the probability distributions of the given patterns belonging to different classes are used to determine the decision boundaries, which must either be specified or learned [103, 112] from the training samples. Discriminant analysis-based approach can also be used for classification: first, a parametric form of the decision boundary (e.g. linear or quadratic) is specified; then the best decision boundary of the specified form is found based on the classification of training patterns. Such boundaries can be constructed using, for example, a mean squared error criterion.

2.1.3.2 Template Based Pattern Recognition

Template matching [63] is one of the simplest and earliest approaches to pattern recognition. Matching is one of the basic operations of pattern recognition, where the similarity between two entities (geometric objects, shapes) is determined, which are of same type.

Template is a prototype of the pattern to be recognized, which is used in template matching. The pattern to be recognized is compared with the stored template, considering all allowable pose (translation and rotation) and scale changes. The similarity measure, used for matching, may be optimized based on the available training set. Sometimes, the template itself can be learned from the available training set. Though template matching is computationally expensive, but with the decrement of cost and availability of faster processors, this technique can now be implemented easily. The technique mentioned above falls under the group of rigid template matching, which is easy to implement and is effective in some application domains, but it has a number of disadvantages. For instance, if the patterns are distorted due to the imaging process, viewpoint change, or large intra-class variations among the patterns, the process of matching fails. Deformable template models [161] or rubber sheet deformations [25] can be used to match patterns when the deformation cannot be easily explained or modeled directly.

2.1.3.3 Syntactic Pattern Recognition

In case of complex pattern recognition problems, sometimes, it is more useful to adopt a hierarchical recognition procedure, where a pattern can be decomposed into some simple sub-patterns which are themselves built from yet simpler sub-patterns [139, 412]. The simplest sub-patterns which can be recognized are called primitives and the actual pattern given is represented in terms of the relationships among the primitives. In case of syntactic pattern recognition [460], a formal analogy is drawn between the structure of patterns and the syntax of a language. The patterns are considered similar to the sentences corresponding to a language, whereas, primitives are considered as the alphabet. The sentences are constructed following a grammar. Hence, a large set of complex patterns can be represented using few primitives and grammatical rules. The grammar for each pattern class must be learned by the use of given training samples. In addition to classification, syntactic pattern recognition also provides the idea how the objects are constructed from the primitives. This method is useful in situations where the structure of the input patterns can be decomposed in terms of a set of rules, like textured images, EKG waveforms, and shape analysis of contours [139]. In case of segmentation [489] of noisy patterns, the implementation of a syntactic approach may leads to difficulties. The notion of attributed grammars, introduced by Fu [139], unifies syntactic and statistical pattern recognition. The syntactic approach sometimes demand large training sets, along with a huge computational power [418].

2.2 Relevance of Soft Computing in Pattern Recognition

The term soft computing automatically brings to mind the opposite concept of hard computing. Hard computing is what we have been used to, what has been going on, right from the beginning of computational science. All the classical reasoning and modeling approaches that are based on Boolean logic, analytical models and crisp classifications fall in this category. Hard computing comes down hard on precision, leaving no room for approximations. This can be computationally expensive, time consuming, and sometimes even impossible for applications related to complex real life problems, because many such problems are typically ill-defined, difficult to model and with large solution spaces. In such cases, soft computing comes to the rescue, but with a price of compromising on some of the principles of hard computing. Soft computing uses inexact way for providing a quick solution to computationally hard tasks, for example, the solution of NP-complete problems, where, there is no known algorithm that can provide an exact solution in polynomial time. Unlike conventional computing, soft computing is tolerant of imprecision, uncertainty, partial truth, and approximation. Soft computing mimics the working principle of human mind. By incorporating the concept of imprecision, uncertainty, partial truth, and approximation, soft computing techniques try to deliver some tractable, robust and low cost solution. The basis of soft computing principles are fuzzy logic [48, 51, 554], neural computing [170, 187], evolutionary computation [352], machine learning [51] and probabilistic reasoning, with the latter subsuming belief networks, chaos theory and parts of learning theory. All the techniques mentioned above have a common property that they are not competitive in nature: in many cases, a complex problem can be solved effectively by using fuzzy logic, neural computing, evolutionary computation, machine learning and probabilistic reasoning when they are combined together.

Fuzzy sets and rough sets are considered as two major building blocks of soft computing. A brief description of fuzzy sets and rough sets is presented in this section along with few other approaches like genetic algorithm and neural networks. This section also includes few hybrid techniques, like rough-neural and rough-GA based approaches.

2.2.1 Fuzzy Sets

Fuzzy set was introduced by Zadeh [554] as a generalization of the classical set theory. To a reasonable extent, fuzzy logic is capable of supporting human type reasoning in natural form. The uncertainty can arise either implicitly or explicitly in each and every phase of a pattern recognition system. It results from the incomplete, imprecise or ambiguous input information, the ill-defined and/or overlapping boundaries among the classes, and the indefiniteness in defining or extracting features and relations among them. Any decision taken at a particular level may have an impact on all other higher level activities. The

modeling of imprecise and qualitative knowledge, as well as the transmission and handling of uncertainty at various stages, is possible through the use of fuzzy sets [51, 253, 331].

A fuzzy set A in a space of objects $X = \{x\}$ is a class of events with a continuum of grades of membership and is characterized by a membership function $\mu_A(x)$ that associates with each element in X a real number in the interval [0, 1] with the value of $\mu_A(x)$ at x representing the grade of membership of x in A. Formally, a fuzzy set A with its finite number of supports $x \in X$ is defined as a collection of ordered pairs $A = \{\mu_A(x)/x, x \in X\}$, where the support of A is an ordinary subset of X and is defined as

$$A = \{x | x \in X \text{ and } \mu_A(x) > 0\}. \tag{2.1}$$

Here, $\mu_A(x)$ represents the degree to which an object x may be a member of A or belong to A. If the support of a fuzzy set is only a single object $x \in X$, then $A = \{\mu_A(x)/x\}$ is called a fuzzy singleton. Hence, if $\mu_A(x) = 1$, $A = 1/x$ denotes a nonfuzzy singleton. In terms of the constituent singletons, the fuzzy set A with its finite number of supports $x \in X$ can also be expressed in union form as

$$A = \bigcup \{\mu_A(x)/x\} \tag{2.2}$$

where the sign $\bigcup$ denotes the union [253]. Assignment of membership functions of a fuzzy subset is subjective in nature, and reflects the context in which the problem is viewed.

The fuzzy set theory has greater flexibility to capture various aspects of incompleteness, impreciseness or imperfection in information about a situation as it is a generalization of the classical set theory [445]. The relevance of fuzzy set theory in the realm of pattern recognition is adequately justified in [51, 238, 391, 556]. Fuzzy sets have been successfully applied in pattern classification [268], clustering [48, 51, 113, 238, 261, 383], and image processing [29, 137, 151, 247, 318, 537]. In addition to pattern recognition, fuzzy sets find widespread applications in solving different problems in association rule mining [21, 306, 527], fuzzy information storage and retrieval [557], functional dependency [59], data summarization [80, 272], web mining [260, 366], granular computing [32, 33], microarray data analysis [41, 98], case-based reasoning [386], and so forth.

2.2.2 Rough Sets

The conventional rough set theory proposed by Pawlak [414] is a very useful technique which can be used to find the data dependencies and hence can be used for pattern recognition applications. Over the past decade, rough set theory has gained high popularity among the researchers. In case of a data set having discretized attributes, rough set theory can find a subset [292, 293] of the original features which are most informative, that is,

most powerful to predict class values. Algorithms based on rough sets do not require any external user specified parameters for the analysis of the data set. The theory of rough sets can be efficiently used in feature selection and clustering applications. In case of feature selection, rough sets can be used to recognize the dispensable and indispensable features using the concept called discernibility matrix [370,577]. In the clustering problem, rough sets divide the samples into two regions for each cluster, where, in one region, the belongingness of the objects in the cluster is certain, and in another region it is uncertain.

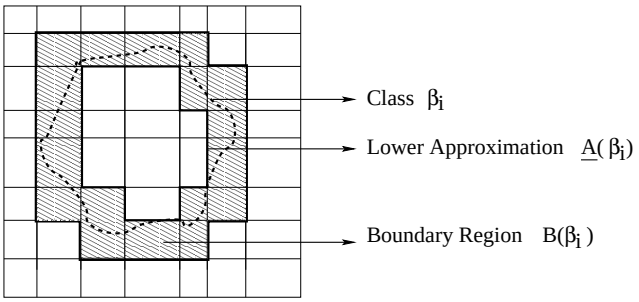


Figure 2.1: Rough Sets

2.2.2.1 Classical Rough Sets

In this subsection, various terms are defined related to rough sets theory (RST).

Definition 2.1 *Let $\mathbb{U}$ be a non-empty set, called as the universe of discourse, and $x, y, z \in \mathbb{U}$. Let $\mathcal{R}$ be a relation, such that, $x\mathcal{R}y$ if and only if $(x, y) \in \mathcal{R}$. $\mathcal{R}$ is called as an equivalence relation, if it satisfies the properties mentioned below:*

1. Reflexive Property: $x\mathcal{R}x$, $\forall x \in \mathbb{U}$.
2. Symmetric Property: $x\mathcal{R}y \Leftrightarrow y\mathcal{R}x$.
3. Transitive Property: $x\mathcal{R}y \wedge y\mathcal{R}z \Rightarrow x\mathcal{R}z$.

Definition 2.2 *Let the partition F be the family of nonempty subsets of $\mathbb{U}$ ($F = \{\mathbb{U}_i\}$), such that, each element of $\mathbb{U}$ is contained in exactly one element of F. Then*

1. $\mathbb{U} = \bigcup_{i=1}^{n} \mathbb{U}_i$,
2. $\mathbb{U}_i \bigcap \mathbb{U}_j = \emptyset$, $\forall (i \neq j)$.

Definition 2.3 *The theory of rough sets is based on the indiscernibility relation. Let $T = \langle \mathbb{U}, \mathbb{A} \rangle$ be a decision system, where $\mathbb{U}$ is a non-empty finite set, called as universe, and*

$\mathbb{A}$ is a set of attributes. $\mathbb{C}$ and $\mathbb{D}$ are two disjoint subsets of $\mathbb{A}$, called as conditional and decisional attributes, respectively. The elements of $\mathbb{U}$ are denoted as samples or objects. For each feature $a \in \mathbb{A}$, there exists a relation $\mathbb{U} \to V_a$, where V_a is called the value domain of a.

Any subset $\mathbb{P}$ of knowledge $\mathbb{A}$ defines an equivalence or indiscernibility relation $IND(\mathbb{P})$ on $\mathbb{U}$:

$$IND(\mathbb{P}) = \{(x, y) \in \mathbb{U} \times \mathbb{U} \mid \forall a \in \mathbb{P},\ a(x) = a(y)\}. \tag{2.3}$$

That means, if $(x, y) \in IND(\mathbb{P})$, we can not discern between them, as they do not differ enough, and so x and y are called indiscernible by the attributes from $\mathbb{P}$.

The partition of $\mathbb{U}$ generated by $IND(\mathbb{P})$ is denoted as

$$\mathbb{U}/IND(\mathbb{P}) = \{[x]_{\mathbb{P}} \mid x \in \mathbb{U}\}; \tag{2.4}$$

where $[x]_{\mathbb{P}}$ is the equivalence class containing x. The elements in $[x]_{\mathbb{P}}$ are indiscernible or equivalent with respect to knowledge $\mathbb{P}$. Equivalence classes, also termed as information granules, are used to characterize arbitrary subsets of $\mathbb{U}$. The equivalence classes corresponding to conditional and decisional attributes are denoted as $\mathbb{U}/IND(\mathbb{C})$ and $\mathbb{U}/IND(\mathbb{D})$ respectively. The equivalence classes of the discernibility relation, which are the minimal blocks of the information system, can be used to approximate these concepts. The equivalence classes of $IND(\mathbb{P})$ and the empty set $\emptyset$ are the elementary sets in the approximation space T.

A set $X \subseteq \mathbb{U}$ could be approximated using lower and upper approximation.

Definition 2.4 *Let $\mathbb{P} \subseteq \mathbb{C}$ and $X \subseteq \mathbb{U}$. The $\mathbb{P}$-lower approximation set of X (denoted as $\underline{\mathbb{P}}X$) is the set of all elements of $\mathbb{U}$ that certainty belong to X.*

$$\underline{\mathbb{P}}X = \{x \in \mathbb{U} \mid [x]_{\mathbb{P}} \subseteq X\}. \tag{2.5}$$

According to the definition, $\underline{\mathbb{P}}X \subseteq X$.

Definition 2.5 *The $\mathbb{P}$-upper approximation set of X (denoted as $\overline{\mathbb{P}}X$) is the set of all elements of $\mathbb{U}$, that may possibly belong to the subset X.*

$$\overline{\mathbb{P}}X = \{x \in \mathbb{U} \mid [x]_{\mathbb{P}} \cap X \neq \emptyset\}. \tag{2.6}$$

Again, according to the definition, $X \subseteq \overline{\mathbb{P}}X$.

Definition 2.6 *Boundary region is a collection of all elements of $\mathbb{U}$ which are included in $\overline{\mathbb{P}}X$ but not in $\underline{\mathbb{P}}X$.*

$$BN(X) = \overline{\mathbb{P}}X - \underline{\mathbb{P}}X. \tag{2.7}$$

Definition 2.7 *A subset X defined in the approximation space $\langle \mathbb{U}, \mathbb{A} \rangle$ through its lower and upper approximations (the tuple $\langle \underline{\mathbb{P}}X, \overline{\mathbb{P}}X \rangle$) is called a Rough set. The set is called a rough set when the boundary region is non-empty, that is, $(\overline{\mathbb{P}}X \neq \underline{\mathbb{P}}X)$.*

Definition 2.8 *A subset is called crisp when its boundary region is empty, that is, $(\overline{\mathbb{P}}X = \underline{\mathbb{P}}X)$.*

Definition 2.9 *It is the set of all objects from the universe $\mathbb{U}$ which can be unambiguously classified to the classes of $\mathbb{U}/\mathbb{D}$ using the attributes $\mathbb{C}$.*

$$POS_{\mathbb{C}}(\mathbb{D}) = \bigcup_{X \subseteq \mathbb{U}/\mathbb{D}} \underline{\mathbb{C}}X; \tag{2.8}$$

where $\underline{\mathbb{C}}X$ denotes the lower approximation of the set X with respect to attributes $\mathbb{C}$. The positive region of the subset X belonging to the partition $\mathbb{U}/\mathbb{D}$ is also called the lower approximation of the set X. The positive region of a decision attribute with respect to the attribute set $\mathbb{C}$ represents approximately the predictive ability of $\mathbb{C}$.

Definition 2.10 *The negative region contains the set of objects that can be definitely ruled out as members of the target set X with respect to the concept $\mathbb{P}$. It consists of all classes that have no overlap with the concept.*

$$NEG_{\mathbb{P}}X = \mathbb{U} - \overline{\mathbb{P}}X. \tag{2.9}$$

Definition 2.11 *Let $\mathbb{U} = \{x_1, x_2, x_3, \cdots, x_n\}$ represents the universe for a decision table. The discernibility matrix is represented as:*

$$m_{ij} = \{a \in \mathbb{C} : (a(x_i) \neq a(x_j)) \wedge (d(x_i) \neq d(x_j), d \in \mathbb{D})\}; \quad \forall i, j = 1, 2, 3, \cdots, n. \tag{2.10}$$

Here, m_{ij} consists of the set of features that classify objects x_i and x_j into two different decision classes in $\mathbb{U}/\mathbb{D}$ partition.

Definition 2.12 *The real life data sets may contain lots of features. Some of these features are very important (indispensable) for the data analysis [31, 93]. The objective of feature selection is to search the indispensable features while eliminating the dispensable features.*

Let $a \in \mathbb{C}$. A feature a is called dispensable in the decision set T if

$$POS_{\{\mathbb{C}-\{a\}\}}(\mathbb{D}) = POS_{\mathbb{C}}(\mathbb{D}); \tag{2.11}$$

otherwise, feature a is called as indispensable, removing which, the predictability power of the resulting condition attribute set will be reduced. The dispensable features can be

eliminated [55] to reduce the dimensionality of the data set.

Definition 2.13 *A reduct is a minimal set of features that preserves the indiscernibility relation produced by a partition of* $\mathbb{C}$*. There can be several such subsets which can have same predictive power as* $\mathbb{C}$*. The attributes other than the reduct are superfluous or irrelevant, so they could be removed without loss of classification performance. A set* $R \in \mathbb{C}$ *is referred as reduct if*

$$POS_R(\mathbb{D}) = POS_{\mathbb{C}}(\mathbb{D}). \tag{2.12}$$

Definition 2.14 *The set of all indispensable attributes in* $\mathbb{C}$ *is represented as* $CORE(\mathbb{C})$*. The core is the set of all single element entries of the discernibility matrix, that is,*

$$CORE(\mathbb{C}) = \{a \in \mathbb{C} | m_{ij} = \{a\}, \forall\ i, j\}. \tag{2.13}$$

Core is also defined as the intersection of all reducts:

$$CORE(\mathbb{C}) = \bigcap\ RED(\mathbb{C}); \tag{2.14}$$

where $RED(\mathbb{C})$ is the set of all reducts in $\mathbb{C}$.

Definition 2.15 *In an information system* $T = \langle \mathbb{U}, \mathbb{A} \rangle$*, where* $\mathbb{A} = \mathbb{C} \cap \mathbb{D}$*,* $\mathbb{C}$ *and* $\mathbb{D}$ *being the sets of condition attributes and decision attributes, respectively, the dependency coefficient between* $\mathbb{C}$ *and* $\mathbb{D}$ *can be defined as*

$$\gamma_{\mathbb{C}}(\mathbb{D}) = \frac{|POS_{\mathbb{C}}(\mathbb{D})|}{|\mathbb{U}|}; \tag{2.15}$$

where $|.|$ *indicates cardinality of a set.*

As $\gamma_{\mathbb{C}}(\mathbb{D})$ expresses the proportion of the objects correctly classified with respect to the total, with the help of the conditional features $\mathbb{C}$, it can take the values between 0 and 1. If γ=1, $\mathbb{D}$ depend totally on $\mathbb{C}$, if $0 < \gamma < 1$, $\mathbb{D}$ depends on $\mathbb{C}$ partially, and if γ=0, then $\mathbb{D}$ is independent on $\mathbb{C}$.

The worst case complexity for finding the γ is $O(n^2 \times m)$, where n is the number of instances and m is the number of attributes.

Definition 2.16 *The accuracy of the approximation for the set* X *is measured as the ratio of the lower and the upper approximation size.*

$$\alpha(X) = \frac{|\underline{X}|}{|\overline{X}|}. \tag{2.16}$$

When there is no boundary region, the ratio is equal to 1, that means the classification is perfect. Thus, a set X is crisp only when the accuracy of approximation is 1, otherwise it is called as rough set.

2.2.2.2 Neighborhood Rough Sets

Given an arbitrary $x_i \in \mathbb{U}$ and $\mathbb{P} \subseteq \mathbb{C}$, the neighborhood $\Phi_{\mathbb{P}}(x_i)$ of x_i with given threshold Φ, in feature space $\mathbb{P}$, is defined as [198]

$$\Phi_{\mathbb{P}}(x_i) = \{x_j | x_j \in \mathbb{U}, \Delta^{\mathbb{P}}(x_i, x_j) \leq \Phi\}; \tag{2.17}$$

where Δ is a distance function. $\Phi_{\mathbb{P}}(x_i)$ in (2.17) is the neighborhood information granule centered with sample x_i. The neighborhood granule generation is effected by two key factors, namely, the used distance function Δ and parameter Φ. The first one determines the shape, while second controls the size of neighborhood granule. Both of these factors play important roles in neighborhood rough sets and can be considered to control the granularity of data analysis. The significance of attributes varies with the granularity levels. Accordingly, neighborhood rough set based algorithm selects different attribute subsets with the change of Δ function and Φ value [315, 392].

Hence, each sample generates granules with a neighborhood relation. For a metric space $\langle \mathbb{U}, \Delta \rangle$, the set of neighborhood granules $\{\Phi(x_i) | x_i \in \mathbb{U}\}$ forms an elemental granule system that covers the universal space rather than partitions it as in case of rough sets. It is noted that the partition of space generated by rough sets can be obtained from neighborhood rough sets with covering principle, while the other way round is not possible [392]. Moreover, a neighborhood granule degrades to an equivalence class for $\Phi = 0$. In this case, the samples in the same neighborhood granule are equivalent to each other and neighborhood rough set model degenerates to rough sets [198].

2.2.3 Relationship Between Fuzzy Sets and Rough Sets

To represent any uncertainty regarding set membership, two different soft computing approaches are mostly used, i) fuzzy sets, introduced by Zadeh [554] in 1965 and ii) rough sets, introduced by Pawlak [413] in 1982.

In fuzzy sets, the membership function is used for assigning a degree of membership to the objects. A fuzzy set A defined on a classical set X is represented as follows:

$$A = \{(x, \mu_A(x)) \mid x \in X\}. \tag{2.18}$$

where, $\mu_A(x)$ represents the fuzzy membership function. A subset A is termed as a fuzzy set when it does not have crisp membership in X, but the memberships are represented using the fuzzy membership function, having the graded values taken from the interval [0,1]. The membership function $\mu_A(x)$ quantifies the grade of membership of the elements x belonging to the set X. The membership value as 0 signifies that the member is not included in the

given set, while 1 represents a fully included member. An object is called as fuzzy member if the membership value is strictly between 0 and 1. Often the concept is represented using a more general definition, where the membership function takes the values using several arbitrary fixed algebra or structure L. This generalized technique was first proposed by Joseph Goguen [154]. The fuzzy set B, where $B = \{(3, 0), (4, 0.2), (5, 0.8), (6, 1), (7, 0.5)\}$ would be enumerated as $B = \{0.2/4, 0.8/5, 1/6, 0.5/7\}$ using standard fuzzy notation. Note that, any object with a zero membership grade is not considered in the set expression. The standard notation to represent the membership grade of the fuzzy set B at 5 is $\mu_B(5) = 0.8$.

On the other hand, in rough sets, the lower and upper approximations for a subset X are generated using the equivalence classes. Rough sets theory is not useful for crisp sets, where in crisp set, the objects do not have any uncertainty about their membership. Rough sets are used to approximate any set that cannot be precisely defined. The rough membership function quantifies the degree or relative overlap between the set X and the equivalence class to which the current argument belongs to.

2.2.4 Rough-Fuzzy Hybridization

The reliance on discrete data of rough set theory for its successful operation is considered as the primary drawback of the approach. In reality, this property is not present in data in general, which is a basic requirement of rough set theory. One can take the example of a medical data set, where some values such as Yes or No cannot be considered as two discrete values for a Headache attribute, as it is not an attribute which can be measured straightforward with a high degree of accuracy. Blood pressure is another real valued attribute. The measurements are consisting of real valued information, but for the purposes of rough sets, it must be discretized into a small set of labels like normal, high, and so forth. Subjective judgments are converted here to some objective measurements. The use of fuzzy-rough sets [10, 11, 110, 114, 219, 222, 419] is a better way to handle this problem, because, in that case, subjective judgments are not entirely removed as fuzzy set membership functions need that for their definition. Moreover, in case of real valued noisy data, the method offers a high degree of flexibility, as it can efficiently model the vagueness and imprecision present in the data.

Similar to the crisp equivalence classes [222], which are the backbone of rough sets, fuzzy equivalence classes are the backbone of the fuzzy-rough approach [111, 494, 548]. In this case, the conditional values and the decision values may all be fuzzy. The concept of fuzzy equivalence classes can be extended from crisp equivalence classes by the inclusion of a fuzzy similarity relation S on the universe, that reflects how similar two elements are. For example, if $\mu_S(x, y) = 0.95$, then elements x and y are considered to be very similar. The properties of equivalence relations, that is, reflexivity ($\mu_S(x, x) = 1$), symmetry ($\mu_S(x, y) =$

$\mu_S(y,x)$), and transitivity ($\mu_S(x,z) \geq \mu_S(x,y) \wedge \mu_S(y,z)$) hold. To model real world situations, transitivity may not always required. For example, let x, y, and z be a large circle, a medium-sized circle, and a small circle. In this case, x and y are more or less similar (for example, their similarity is 0.6). It may also be the case that y and z are more or less similar to each other (for example, their similarity is 0.6). But, x and z may not be similar (for example, their similarity is 0.15), and hence transitivity does not hold. But the transitive properties of fuzzy relations are often desirable from mathematical point of view.

Definition 2.17 *Using the fuzzy similarity relation, the fuzzy equivalence class $[x]_S$ for objects similar to x can be defined:*

$$\mu_{[x]_S}(y) = \mu_S(x,y). \tag{2.19}$$

The following axioms should hold for a fuzzy equivalence class F [192]:

1. $\exists x$, $\mu_F(x) = 1$ (μ_F is normalized)
2. $\mu_F(x) \wedge \mu_S(x,y) \leq \mu_F(y)$
3. $\mu_F(x) \wedge \mu_F(y) \leq \mu_S(x,y)$

The first axiom states that an equivalence class is nonempty. The second axiom represents that elements in y's neighborhood are in the equivalence class of y. The last axiom states that any two elements in F are related via S. Clearly, this definition degenerates to the classical definition of equivalence classes when S is non-fuzzy (crisp). The family of normal fuzzy sets produced by a fuzzy partitioning of the universe of discourse can play the role of fuzzy equivalence classes [111]. Let us take the crisp partitioning of a universe of discourse, $\mathbb{U}$, by the attributes in $Q : \mathbb{U}/Q = \{\{1,4,5\},\{2,3,6\}\}$. This contains two equivalence classes ($\{1,4,5\}$ and $\{2,3,6\}$) that can be thought of as degenerated fuzzy sets, with those elements belonging to the class possessing a membership of one, and zero otherwise. For example, for the first class, the objects 2, 3, and 6 have a membership of zero. This technique can be extended to the case of fuzzy equivalence classes in a straightforward way, where, with respect to any given class, objects can have membership values between the interval [0, 1]. Here, $\mathbb{U}/Q$ is not restricted to crisp partitions only, but it can create fuzzy partitions [336] also.

2.2.4.1 Fuzzy-Rough Sets

The definition of fuzzy-rough sets given by Dubois and Prade [111] are originated from Waillaeys and Malvache [513] for defining a fuzzy set with respect to a family of fuzzy

sets. It deals with the approximation of fuzzy sets in a fuzzy approximation space defined by a fuzzy similarity relation or by a fuzzy partition. In this subsection, fuzzy-rough sets are defined based on a fuzzy similarity relation S, which is a fuzzy subset of $(\mathbb{U} \times \mathbb{U})$ and has following three properties:

1. Reflexive Property: $\forall x \in \mathbb{U},\ \mu_S(x, x) = 1$
2. Symmetry Property: $\forall x, y \in \mathbb{U},\ \mu_S(x, y) = \mu_S(y, x)$
3. Transitivity Property: $\forall x, y, z \in \mathbb{U},\ \mu_S(x, z) \geq \min[\mu_S(x, y),\ \mu_S(y, z)]$

For any fuzzy similarity relation S, a fuzzy approximation space can be defined by the pair $\langle \mathbb{U}, S \rangle$. A fuzzy equivalence class $[x]_S$ is a fuzzy partition of the universe that can be constructed by a fuzzy similarity relation, which is a group of elements close to x and defined by (2.20).

$$\mu_{[x]_S}(y) = \mu_S(x, y). \tag{2.20}$$

$\mathbb{U}/S$ represents the family of all fuzzy equivalence classes. For a fuzzy set, its approximation in the fuzzy approximation space is called a fuzzy-rough set, which is represented as a pair of fuzzy sets on $\mathbb{U}/S$.

Definition 2.18 *The fuzzy $\mathbb{P}$-lower and $\mathbb{P}$-upper approximations are defined as [111]*

$$\begin{aligned} \mu_{\underline{\mathbb{P}}X}(F_i) &= \inf_x \max\{1 - \mu_{F_i}(x), \mu_X(x)\}, \quad \forall i; \\ \mu_{\overline{\mathbb{P}}X}(F_i) &= \sup_x \min\{\mu_{F_i}(x), \mu_X(x)\}, \quad \forall i; \end{aligned} \tag{2.21}$$

where F_i denotes a fuzzy equivalence class belonging to $\mathbb{U}/\mathbb{P}$, the partition of $\mathbb{U}$ generated by $\mathbb{P}$, and $\mu_X(x)$ represents the membership of x in X.

These definitions diverge a little from the crisp upper and lower approximations, as the memberships of individual objects to the approximations are not explicitly available.

Definition 2.19 *The fuzzy lower and upper approximations are redefined here as*

$$\begin{aligned} \mu_{\underline{\mathbb{P}}X}(x) &= \sup_{F_i \in \mathbb{U}/\mathbb{P}} \min\{\mu_{F_i}(x), \mu_{\underline{\mathbb{P}}X}(F_i)\}; \\ \mu_{\overline{\mathbb{P}}X}(x) &= \sup_{F_i \in \mathbb{U}/\mathbb{P}} \min\{\mu_{F_i}(x), \mu_{\overline{\mathbb{P}}X}(F_i)\}. \end{aligned} \tag{2.22}$$

Combining (2.21) and (2.22),

$$\mu_{\underline{\mathbb{P}}X}(x) = \sup_{F_i \in \mathbb{U}/\mathbb{P}} \min[\mu_{F_i}(x), \inf_y \max\{1 - \mu_{F_i}(y), \mu_X(y)\}];$$
$$\mu_{\overline{\mathbb{P}}X}(x) = \sup_{F_i \in \mathbb{U}/\mathbb{P}} \min[\mu_F(x), \sup_y \min\{\mu_{F_i}(y), \mu_X(y)\}]. \tag{2.23}$$

In implementation, only those $y \in \mathbb{U}$ are considered where $\mu_{F_i}(y)$ is nonzero, that means, where y is a fuzzy member of fuzzy equivalence class F_i. The tuple $\langle \underline{\mathbb{P}}X, \overline{\mathbb{P}}X \rangle$ is called a fuzzy-rough set.

These definitions degenerate to traditional rough sets when all equivalence classes are crisp. The crisp lower approximation is:

$$\mu_{\underline{\mathbb{P}}X}(x) = \begin{cases} 1, & x \in F_i, F_i \subseteq X \\ 0, & \text{otherwise} \end{cases} \tag{2.24}$$

which means, if an object x belongs to an equivalence class that is a subset of X, then it belongs to the $\mathbb{P}$-lower approximation of X. Again, the fuzzy lower approximation must behave in exactly the same way as of the crisp definition for crisp situations. Fuzzy lower approximation (2.23) can be rewritten as

$$\mu_{\underline{\mathbb{P}}X}(x) = \sup_{F_i \in \mathbb{U}/\mathbb{P}} \min[\mu_{F_i}(x), \inf_y \{\mu_{F_i}(y) \rightarrow \mu_X(y)\}]; \tag{2.25}$$

where $\rightarrow$ indicates the fuzzy implication, using the traditional min-max interpretation. $\mu_{F_i}(x)$ and $\mu_X(x)$ will take values from $\{0, 1\}$ in the crisp scenario. So $\mu_{\underline{\mathbb{P}}X}(x)$ only takes zero value when at least one object in its equivalence class F_i fully belongs to F_i but not to X, which matches with the definition of the crisp lower approximation. In the similar way, the specialization from fuzzy $\mathbb{P}$-upper approximation can be shown.

2.2.4.2 Rough-Fuzzy Sets

It is also defined by Dubois and Prade by approximating fuzzy sets in an approximation space [478], which is actually a special case of fuzzy-rough sets. A rough-fuzzy set is a generalization of rough set, and is derived from the approximation of a fuzzy set in a crisp approximation space. In case of rough-fuzzy set, the decision attributes are fuzzy and condition attributes are crisp.

Definition 2.20 *Given a fuzzy set X, the result of approximation is a pair of fuzzy sets defined on the equivalence class $\mathbb{U}/IND(\mathbb{P})$, called the lower and upper approximations,*

which are defined as

$$\mu_{\underline{\mathbb{P}}X}(F_i) = \inf_x\{\mu_X(x) \mid x \in F_i\}, \quad \forall i;$$
$$\mu_{\overline{\mathbb{P}}X}(F_i) = \sup_x\{\mu_X(x) \mid x \in F_i\}, \quad \forall i; \tag{2.26}$$

where F_i represents a fuzzy equivalence class belonging to $\mathbb{U}/\mathbb{P}$, the partition of $\mathbb{U}$ generated by $\mathbb{P}$, and $\mu_X(x)$ represents the membership of x in X.

Definition 2.21 *Using the extension principle, the pair can be extended to a pair of rough sets on the universe $\mathbb{U}$ as defined by (2.27).*

$$\mu_{\underline{\mathbb{P}}X}(x) = \inf_x\{\mu_X(x) \mid x \in F_i\}, \quad \forall i;$$
$$\mu_{\overline{\mathbb{P}}X}(x) = \sup_x\{\mu_X(x) \mid x \in F_i\}, \quad \forall i. \tag{2.27}$$

The pair $\langle \underline{\mathbb{P}}X, \overline{\mathbb{P}}X \rangle$ is called a rough-fuzzy set on $\mathbb{U}$ with reference fuzzy set X. Fuzzy-rough sets [111], where all equivalence classes may be fuzzy, is the generalization of rough-fuzzy sets. So rough-fuzzy sets is useful where condition attributes are crisp and decision attributes are fuzzy, where as, fuzzy-rough sets can be applied for analysing the data sets where both the condition and decision attributes may be either fuzzy or crisp.

2.2.4.3 Other Rough-Fuzzy Hybridization

Different hybrid techniques are also available in [415] other than the fuzzy-rough definitions given before. The concepts of information theoretic measures are related to rough sets [36] to establish the rough set models of uncertainty. This technique is based on an alternative definition of fuzzy-rough sets, originating from the rough membership function [414], and has been applied to some rough and fuzzy-rough relational database models. In [536], rough set is represented by a fuzzy membership function to define the positive, boundary, and negative regions. The objects in the positive region are given the membership as 1, where the objects, those belong to the boundary region, get a membership value as 0.5. The objects belonging to the negative region have zero membership. In this way, a rough set is represented using a fuzzy set, with required modifications in the union and intersection operations. The reason behind introducing fuzziness into rough sets is to quantify the levels of roughness in the boundary region by using fuzzy membership values. So, instead of just assigning the membership value as 0.5 for the boundary region, it is better to allow the elements to have any membership values from the range of [0, 1]. Hence, using this approach, a fuzzy-rough set Y can be defined using a membership function $\mu_Y(x)$ that can assign a grade of membership (in the interval [0, 1]) to every element of $\mathbb{U}$. In case of

a rough set X with a crisp equivalence relation R, the following relations hold:

$$\begin{aligned} \mu_Y(\underline{R}X) &= 1; \\ \mu_Y(\mathbb{U} - \overline{R}X) &= 0; \\ 0 < \mu_Y(\overline{R}X - \underline{R}X) &< 1. \end{aligned} \tag{2.28}$$

The technique mentioned above cannot be considered as true hybridization of the two approaches, as the method is only assigning a degree of membership to the elements depending on the crisp positive, boundary, or negative region they belong to. The concept of fuzzy equivalence classes are not utilized, and so this approach is not really useful for fuzzy-rough attribute reduction.

In [415], a different method has been proposed, which blurs the distinction between rough and fuzzy sets. The research was fueled by the concern that a purely numeric fuzzy set representation may be too precise; a concept is described exactly once its membership function has been defined. But in this approach, excessive precision is required for describing some imprecise concept.

2.2.5 Other Approaches of Soft Computing

Few other approaches of soft computing are reported next, including some hybrid approaches.

2.2.5.1 Genetic Algorithm

John Holland first used the term genetic algorithm [352], abbreviated to GA. Many other scientists also worked on the same idea in parallel. In Germany, Ingo Rechenberg and Hans-Paul Schwefel developed the idea of the evolutions strategie [47] (in English, evolution strategy) in 1960. In the same year, Bremermann, Fogel and others in the USA implemented the idea for what they called evolutionary programming [128]. The common among these ideas is the use of mutation and selection. However, Holland influenced the most. He introduced the idea of recombination for mutation in GA. Moreover, his idea only could provide the full capabilities of GA in terms of optimization. The concept of GA is based on the philosophy of natural selection or survival of the fittest (Darwin 1859) and genetics which are inspired by biological structures and their evolution. The use of GA is very much effective while the search space is large, complex and multimodal. Unlike traditional search techniques, it works with the coding of problem variables instead of variable themselves. Moreover, it can search simultaneously from multiple points and this fact makes GA parallel in nature. This also helps increasing the probability of avoiding the issue of getting trapped into local optimal solution. According to the GA, the deci-

sion variables of the search problem are encoded into a finite length string of symbols of certain cardinality. The symbols are called genes and the values of genes are called alleles, whereas, the string is referred as chromosome. GA starts with the random population of chromosomes. Each chromosome is evaluated based on some defined fitness value and then takes part into selection process, where the chromosomes having better fitness are given more chance to reproduce than the others. Thereafter, crossover and mutation operators are applied on the selected chromosomes. This preserves the important information and helps to achieve the good solution for next generation. This process continues until some termination condition is reached. Goldbergs [155] pseudo-code of genetic algorithms is outlined in Algorithm 1.

Input: Initial Population $P^{(0)}$

Output: Optimum Solution $Best\{P^{(t)}\}$

$t = 0$
Initialize $P^{(t)}$
Evaluate $P^{(t)}$
repeat
  $t = t + 1$
  Select $P^{(t)}$ from $P^{(t-1)}$ `/* Selection process */`
  Recombine pairs in $P^{(t)}$ `/* Cross-over process */`
  Mutate $P^{(t)}$ `/* Mutation process */`
  Preserve Elites in $P^{(t)}$ `/* Elitism */`
  Evaluate $P^{(t)}$
until (*Terminating Condition*)

Algorithm 1: Goldbergs Pseudo-code of Genetic Algorithms

In Algorithm 1, t and P stand for generation number and population respectively. Initialization process initialize the population and the fitness is computed in evaluation process. The major steps of GA are described as follows.

Initial Population: Generation of an initial population is the first step of a GA. Each individual of the population encodes a possible solution to a problem. The population size is dependent on the problem domain, but generally consists of several hundreds or thousands of possible solutions. Often, the initial population is generated randomly, which allows the possible solutions coming from the entire range. When the precise domain knowledge is available, the solutions may be "seeded" in areas where the possibility of getting the optimal solutions is very high.

Fitness Evaluation: After creating the initial population, each individual is evaluated and assigned a fitness value according to the fitness function. Genetic Algorithm can automatically search an optimum solution according to fitness evaluation. Each individual of the population, therefore, needs to be awarded a figure of merit, to indicate how close it came to reach the optimum, and this is generated by applying the fitness function to the test, or simulation, results obtained from that solution.

Selection Process: During selection process, chromosomes from the parent populations are selected based on the fitness values and an intermediate population, called mating pool is maintained. These selected chromosomes will further take part into subsequent processes like crossover and mutation. Three selection methods, namely, roulette wheel selection [401], stochastic universal sampling [56] and binary tournament selection [56] are very much popular and mostly used. According to roulette wheel selection methods, the wheel has the slots same as the number of population size and the area of each slot is the fitness of that particular chromosome in the population. The wheel is let spinning and the chromosome is selected according to the position of the marker while wheel stops. Thus how many times a particular chromosome is selected, depends on its fitness in the populations, because only fitness determines the area of the slot for that chromosome.

Crossover Process: In this process, two parent chromosomes produce two new chromosomes called offspring. Offspring chromosomes might get the best characteristics from both the parents and become better than their parents. Crossover does not occur all the time. It occurs based on some user defined probability. There are three widely used categories, single point crossover, two-point and uniform crossover. According to single point crossover, crossover process selects a crossover point over a chromosome randomly. Thereafter, it interchanges the two parent chromosomes at this point to produce offspring chromosomes.

Mutation Process: Mutation is another genetic operator. It alters one or more genes in a chromosome and can produce entirely a new chromosome than its initial state. This new chromosome is added in population with the assumption that the new chromosome might give the better solution. This process also follows some user defined probability for occurrence. The probability of mutation is normally kept less (as low as 0.01) than crossover.

Elitism: During the GA process, there is a chance of loosing the best solution in next generation, unless we store the best solution so far in a safe place. The process which ensures it is called elitism.

Genetic algorithm has been successfully applied in pattern classification [7, 39, 99, 225, 250, 270, 303, 371, 423, 475], clustering [6, 38, 82, 107, 148, 172, 176, 230], and feature selection [7, 88, 122, 228, 301, 303, 373, 403, 515].

2.2.5.2 Neural Network

Neural networks can be visualized as parallel computing systems having massive computing power and composed using a huge number of simple processors with lots of interconnections between them. Neural network tries to implement several properties (such as learning, adaptivity, generalization, distributed representation, computation, fault tolerance, and so on) in a network. It uses weighted directed graphs where artificial neurons act as nodes having weighted directed edges, which connect neuron outputs with neuron inputs. Neural networks are able to learn complex nonlinear relationships between inputs and outputs successfully using sequential training procedures, and apply them for various pattern recognition purposes. Feed-forward network is the most well-known class of neural networks used for pattern classification [217], that uses multilayer perceptron and radial-basis function (RBF). Multiple layers are present having few unidirectional connections between the layers. Another common network, generally used in case of object clustering and feature mapping, is the self-organizing map (SOM), also known as Kohonen-network [256]. The learning phase actually updates the network architecture and interconnection weights in such a way that a specific classification/clustering task can easily be performed by a network. Neural networks are gaining increased popularity as they are able solve pattern recognition problems efficiently because of their seemingly low dependency on domain-specific information (compared to rule-based and model-based techniques) and the readiness of efficient learning algorithms for the users. Various robust nonlinear algorithms are designed using neural networks for feature extraction (using hidden layers) and classification (such as, multilayer perceptrons). The traditional feature extraction and classification algorithms can also be implemented using neural network to decrease the hardware cost. Though the underlying principles are widely different, most of the common neural network techniques can be mapped with classical statistical pattern recognition techniques. The relationship between neural networks and statistical pattern recognition is also discussed by Ripley [439] and Anderson et al. [14]. Anderson et al. shown that neural networks are statistics for amateurs, where the statistics are hidden from the user. Along with this, neural networks also provide unified approaches for feature extraction and classification, and can be used to find good, moderately nonlinear solutions.

Neural networks have been successfully applied in multiple areas of pattern recognition [227, 451, 521] and others, like signature verification [170], feature extraction [574], classification [225, 250], demand forecasting [228], fault analysis [186], image classifica-

tion [4, 475], etc.

2.2.5.3 Rough-Neural Approach

Neural networks are able to solve complex problems easily. However, they do not have the ability to separate the redundant information from huge real life data sets, which leads to the problems like long training time, too complex network structure, excessive computation, low converging speed, and so on. a A hybrid technique, called rough-neural approach [106, 542] has been introduced by Hassanien and Slezak [183], which can be used for rule generation and image classification. The basic study consist of only one input layer, one hidden layer, and one output layer. The inputs coming from the external environment is accepted by input layer neurons. The outputs coming from the input layer neurons are send to the hidden layer neurons. The hidden layer neurons propagate their output to the output neurons, which are connected to the external environment.

In contrary of a conventional neuron which is having a single valued output, the output of a rough neuron is a pair of upper and lower bounds. Lingras [286] introduced the concept of rough neuron in 1996. It was defined using lower bound (L_n) and upper bound (U_n), and inputs were assessed relative to boundary values. Rough neuron possess three different types of connections: input-output connection to L_n, input-output connection to U_n, and interconnection between U_n and L_n. Rough-neural approach has been successfully applied in various fields [4, 186].

2.2.5.4 Rough-GA Approach

Lingras [288] proposed a rough sets based classification technique using genetic algorithms. In case of unsupervised learning, this rough sets based genetic encoding is useful in general. A rough sets genome consists of upper and lower bounds for sets in a partition. It can generate partitions like the conventional classes or a more general classification scheme may be provided. A complete description is given in the paper [288], how to design and implement using rough sets genomes. The algorithm provided the best classification accuracy when compared with other unsupervised schemes.

2.3 Brief Survey on Dimensionality Reduction

Dimension reduction [43, 68, 231, 254] is aimed at identifying a minimal-sized subset of attributes (features) that are relevant to the target concept [91]. The objective of dimensionality reduction is three-fold: improving the prediction accuracy, providing faster and more cost-effective prediction, and providing a better understanding of the underlying process that generated the data [169]. There are two main methods for dimension reduction:

feature selection [175, 493] and feature extraction [112, 493, 574]. Few of the commonly used methods for feature selection and feature extraction are discussed below.

2.3.1 Selection-Based Reduction

The problem of feature selection is defined as follows: given a set of m features, select a subset of size d, which leads to the smallest classification error. The relationship between a feature selection algorithm and the measure chosen to evaluate the usefulness of the feature selection process can be of three main forms: filter, wrapper, and embedded.

2.3.1.1 Filter Methods

When the process of feature selection takes place before the induction step, the process is known as filter approach, as it filters the non-useful features prior to induction. Filter methods evaluate the efficiency of the candidate feature subset with respect to the intrinsic characteristic of the data only, based on the relationship of each single feature or in combination with the class label by using some simple statistical measures computed from the empirical distribution [91,93,173]. The simplest way is ranking the features with respect to some univariate scoring metric. Then, the top d features are selected according to their score. Among the large variety of measures found in literature, the most common types are distance or probabilistic metrics and measures inspired by the information theory. Moreover, the filter methods are independent of the induction technique.

2.3.1.2 Wrapper Methods

Wrapper approach [254] finds the subset of features in the feature space and evaluate the performance of the different feature subsets found by calculating the accuracy percentage with respect to some specific classifier, where the training of the classifier is also performed with the respective feature subset. In this technique, the feature selection algorithm uses the learning algorithm as a sub-routine [231]. The general argument in favor of this scheme is to equal the bias of both the feature selection algorithm and the learning algorithm that will be used later on to assess the goodness of the solution. The technique has huge computational overhead because of the induction algorithm that is used to evaluate each candidate subset of features, which is the main disadvantage of this technique. Another issue is, in most of the cases, the feature subset selected using this method becomes biased on the classifier used. Therefore it results in poor performance when different classifier is used.

2.3.1.3 Embedded Methods

Embedded methods combine the qualities of filter and wrapper methods. In this scheme, the classifier or inducer uses its own built-in feature selection algorithm implicitly or explicitly. The methods to induce logical conjunctions [528] provide an example of this embedding. Some popular examples of these methods are LASSO [495] and RIDGE [191] regression which uses its in-built penalization functions for reducing over-fitting. Several traditional machine learning tools like decision trees or artificial neural networks are also included in this scheme [353].

2.3.2 Transformation-Based Reduction

Transformation based reduction techniques are commonly known as feature extraction, that determine an appropriate subspace of dimension d (either in a linear or a nonlinear way) from the original feature set of dimension m ($d < m$). Linear transforms, such as principal component analysis (PCA) [233], independent component analysis (ICA) [43, 68, 273], linear discriminant analysis (LDA) [141], and projection pursuit [136], have been widely used in pattern recognition for feature extraction and dimensionality reduction. There are several ways to define nonlinear feature extraction techniques. One such method, which is directly related to PCA, is called the kernel PCA [184, 461]. Multidimensional scaling is another nonlinear feature extraction technique [58, 359, 455]. Neural networks are also used for feature extraction [86, 142, 256, 299].

Depending on the data and the transformation process, the reduction techniques are classified into different categories. In following subsections, we deal with these different techniques. Generally, the methods are divided into two categories, namely, linear and non-linear methods.

2.3.2.1 Linear Methods

Data which have linear relationship is called as linear data and others are called as non linear data. There are a number of techniques available to handle this type of linear data. Linear discriminant analysis (LDA) [141, 210] and principal component analysis (PCA) [233] fall under this domain.

2.3.2.2 Nonlinear Methods

In real world, most of the data are in the form of non linear. Handling these types of data for further analysis is difficult. There are many techniques, that can handle this type of non linear data. Support vector machine (SVM) [479, 569], independent component analysis (ICA) [43, 68, 208, 273], and multi dimensional scaling (MDS) [85] typically fall under this domain.

2.3.3 Dimensionality Reduction Using Fuzzy-Rough Sets

This section presents different methods for dimensionality reduction using fuzzy-rough sets, which are mostly used and found in the literature. Fuzzy-rough sets [222] encapsulate the related, but distinct concepts of vagueness (for fuzzy sets [554]) and indiscernibility (for rough sets), both of which occur as a result of uncertainty in knowledge [111]. A fuzzy-rough set is defined using two fuzzy sets, fuzzy lower and upper approximations, that are obtained by extending the corresponding conventional rough set notions. In the conventional case, elements that belong to the lower approximation are said to belong to the approximated set with absolute certainty, that means, they have a membership of 1. In the fuzzy-rough variant, elements may have any membership value in the range of [0, 1], allowing greater flexibility in handling uncertainty.

Fuzzy-rough feature selection (FRFS) [221] provides a means by which discrete or real valued noisy data (or a mixture of both) can be effectively reduced without the need for user-supplied information. Moreover, this technique can be applied to data with continuous or nominal decision attributes, and can be applied to both, regression as well as classification data sets. The only additional information required is in the form of fuzzy partitions for each feature that can be automatically derived from the data. This avoids the need for domain experts to provide information on the data involved and also uses the advantage of rough sets in that it requires no information other than the data. However, when such experts are readily available, the knowledge from them can be gathered in the form of fuzzy data partitions, which improves the transparency of the selection process and any other future processes (for example, rule induction).

FRFS uses the concept of fuzzy lower approximation to reduce the real valued features from a given data set. The process is similar to the crisp approach when dealing with nominal well-defined features. The crisp positive region in traditional rough set theory is defined as the union of the lower approximations. By the extension principle [555], the membership of an object $x \in \mathbb{U}$, belonging to the fuzzy positive region can be defined by

$$\mu_{POS_{\mathbb{P}}(Q)}(x) = \sup_{X \in \mathbb{U}/Q} \mu_{\underline{\mathbb{P}}X}(x). \tag{2.29}$$

Object x will not belong to the positive region only if the equivalence class it belongs to is not a constituent of the positive region. This is equivalent to the crisp version, where, objects belong to the positive region only if their underlying equivalence class does so. In the similar way, the negative and boundary regions can be defined. In this feature selection method, the upper approximation is not used.

Using the definition of the fuzzy positive region, the dependency function is defined as

follows:

$$\gamma_{\mathbb{P}}(Q) = \frac{|\mu_{POS_{\mathbb{P}}(Q)}(x)|}{|\mathbb{U}|} = \frac{\sum_{x \in \mathbb{U}} \mu_{POS_{\mathbb{P}}(Q)}(x)}{|\mathbb{U}|}. \tag{2.30}$$

In case of rough sets, the dependency of Q on $\mathbb{P}$ is the proportion of objects that are discernible out of the entire data set. In the fuzzy-rough approach, this is calculated as the fuzzy cardinality of $\mu_{POS_{\mathbb{P}}(Q)}(x)$, divided by the total number of objects in the universe. The definition of dependency degree covers the crisp case as its specific instance. If a function $\mu_{POS_{\mathbb{P}}(Q)}(x)$ is defined in such a way that returns 1 when the object x belongs to the positive region, 0 otherwise, then the above definition may be rewritten as

$$\gamma_{\mathbb{P}}(Q) = \frac{|POS_{\mathbb{P}}(Q)|}{|\mathbb{U}|}; \tag{2.31}$$

which is identical to (2.15). The fuzzy-rough attribute reduction process should find out the dependency among various subsets of the original feature set. In case of two condition attributes, the degree of dependency of the decision feature(s) with respect to $\mathbb{P} = \{a, b\}$ must be determined. In the rough case, $\mathbb{U}/\mathbb{P}$ contains sets of objects grouped together that are indiscernible according to both features a and b. In the rough-fuzzy case, objects may belong to many equivalence classes, so the Cartesian product of $\mathbb{U}/IND(a)$ and $\mathbb{U}/IND(b)$ must be considered in determining $\mathbb{U}/\mathbb{P}$. In general,

$$\mathbb{U}/\mathbb{P} = \otimes\{\mathbb{U}/IND(a) \mid a \in \mathbb{P}\}; \tag{2.32}$$

where

$$A \otimes B = \{X \cap Y \mid X \in A, Y \in B, X \cap Y = \emptyset\}. \tag{2.33}$$

Each set in $\mathbb{U}/P$ denotes an equivalence class. For example, if $\mathbb{P} = \{a, b\}$, $\mathbb{U}/IND(a) = \{N_a, Z_a\}$ and $\mathbb{U}/IND(b) = \{N_b, Z_b\}$, then

$$\mathbb{U}/\mathbb{P} = \{N_a \cap N_b, N_a \cap Z_b, Z_a \cap N_b, Z_a \cap Z_b\}. \tag{2.34}$$

The extent to which an object belongs to such an equivalence class is therefore calculated by using the conjunction of constituent fuzzy equivalence classes, say F_i, $i = 1, 2, \cdots, n$:

$$\mu_{F_1 \cap \cdots \cap F_n}(x) = \min[\mu_{F_1}(x), \mu_{F_2}(x), \cdots, \mu_{F_n}(x)]. \tag{2.35}$$

2.3.3.1 Fuzzy-Rough Max-Dependency

In case of traditional rough sets based attribute reduction [221], a reduct is defined as a subset R of the features that have the same information content as the full feature set C. So, practically this means that the values $\gamma_R(D)$ and $\gamma_C(D)$ are identical (and equal to 1 if the data set is consistent). However, in the fuzzy-rough approach, this situation will not happen, as the uncertainty, encountered when objects belong to many fuzzy equivalence classes, results in a reduced total dependency. Here, the degree of dependency of a set of decision features D can be determined upon the full feature set and can be used as the denominator rather than $|\mathbb{U}|$ (for normalization), allowing γ to reach 1. The features are combined and selected one at a time and the process terminates when the addition of any remaining feature does not increase the dependency.

Input: C, the set of all conditional features; D, the set of decision features

Output: R, the reduced feature subset

$R \leftarrow \{\}$; $\gamma_{best} = 0$; $\gamma_{prev} = 0$
while $\gamma_{best} \neq \gamma_{prev}$ **do**
  $T \leftarrow R$
  $\gamma_{prev} = \gamma_{best}$
  for ***each*** $x \in (C - R)$ **do**
    if $\gamma_{R \cup \{x\}}(D) > \gamma_T(D)$ **then**
      $T \leftarrow R \cup \{x\}$
      $\gamma_{best} = \gamma_T(D)$
    end
  end
  $R \leftarrow T$
end
return R

Algorithm 2: Fuzzy-Rough Max-Dependency Algorithm

As the degree of dependency measure is non-monotonic, it is possible that the Quick-Reduct [83] style search terminates having reached only a local optimum, where the global optimum may lie elsewhere in the search space. However, the algorithm as presented (Algorithm 2) is still highly useful in locating good subsets quickly, but the complexity is high as it adds one feature at a time in the selected subset and calculates the combined dependency, where the evaluation is needed for all the feature subsets, created by adding one unselected feature with the already selected features at a time.

Another way of selection may be done reversing the search process, that is, start with the full set of features and gradually remove the least informative features. This process is repeated until no more features can be removed without reducing the total dependency.

Again, this technique is not suitable for high dimensional data sets as the cost of evaluating these larger feature subsets is too high.

2.3.3.2 Fuzzy-Rough Max-Relevance

As Max-Dependency algorithm has high computational complexity, an alternative approach is to select features based on maximal relevance or Max-Relevance criterion [315]. Max-Relevance algorithm searches a set of features that approximates Max-Dependency criterion with the mean value of all dependency values between individual feature and target class label. The fuzzy-rough based max-relevance (FRMR) is presented in Algorithm 3.

Input: C, the set of all conditional features; D, the set of decision features, S, required number of features

Output: R, the reduced feature subset

$R \leftarrow \{\}$
while $|R| < S$ **do**
  $R \leftarrow R \cup \arg \max_{x \in (C-R)} \{\gamma_x(D)\}$
end
return R

Algorithm 3: Fuzzy-Rough Max-Relevance Algorithm

An attribute selection method was presented in [327], which is based on rough sets and max-relevance max-significance (MRMS) criterion. In [315], a rough-fuzzy MRMS based feature selection method was proposed, which was further improved in [316] using IT2 rough-fuzzy based approach. A rough-fuzzy based simultaneous attribute selection and feature extraction method was proposed in [314].

2.4 Brief Survey on Clustering

In most of the pattern recognition applications, it is really difficult, or sometimes impossible, to put the unlabeled objects into proper groups. The objective of clustering is to find out the optimum decision boundaries for the grouping. Clustering is often called as unsupervised classification which provides a family of procedures that can be used to find out the natural groups present among the patterns using the similarities measured in multi-dimensional space. Numerous clustering algorithms are being emerged day-by-day with increased accuracy and speed [52, 69, 138, 235], which can handle large data set easily. So it is difficult to put them in a specific category as they may have features similar to the

methods of different categories. This subsection presents a brief introduction to different clustering algorithms.

2.4.1 Hierarchical Methods

Here the data objects are hierarchically decomposed [112, 493] into certain groups. Based on the direction of decomposition, the hierarchical method is classified into several categories. Examples of hierarchical clustering algorithms are Chameleon [242], ROCK [165], CURE [165], BIRCH [165], etc.

2.4.1.1 Agglomerative

Agglomerative approach [112,493] (for example, AGNES [95,368]) is a bottom up approach which initially considers each objects as a separate group, and then merges the groups successively until some terminating criteria is satisfied.

2.4.1.2 Divisive

Divisive approach [459] is a top down one, which initially assumes all the objects in a same cluster and splits up into smaller clusters in each iteration until some termination condition is satisfied.

2.4.1.3 Single-Link Clustering

In case of single-link or single-linkage [441] clustering, at each step, the algorithm merges the two clusters where resulting cluster is having the smallest distance, that is, the two clusters having the smallest minimum pairwise distance.

2.4.1.4 Complete-Link Clustering

In case of complete-link or complete-linkage [67] clustering, at each step, the two clusters are merged where the resulting cluster has the smallest diameter, that is, the two clusters with the smallest maximum pairwise distance.

2.4.1.5 Average-Link Clustering

Average-link or group-average [443] clustering is a compromise between the sensitivity of complete-link clustering to outliers and the tendency of single-link clustering to form long chains that do not correspond to the intuitive notion of clusters as compact, spherical objects.

2.4.2 Partitioning Methods

The partitioning method [443] generates k-partitions out of n ($k < n$) objects in the data set, where each partition represents a cluster. The resulting k clusters together satisfy the following criteria: (i) each group must contain one object in minimum. (ii) each sample must be the member of exactly one group only. The scheme is called good partitioning, where the objects in a same cluster are as close as possible, where as, the objects of different clusters are far apart. The distance can be measured in various ways. Examples of partitioning-based clustering algorithms are the k-means algorithm [297,305], k-modes [204] method, k-mediods (for example, PAM, CLARA [204], CLARANS [124, 368]) algorithm.

2.4.2.1 Error Minimization Algorithms

This category is considered as most frequently used methods, which works best in case of isolated and compact clusters. The basic idea behind this is to find a clustering solution by minimizing some error criterion which measure the "distance" of each instance from its representative point. The sum of squared error (SSE) [112, 216] is the most common one, which measures the total squared Euclidian distance of samples to their representative points.

2.4.2.2 Graph-Theoretic Clustering

This method produces clusters via graphs. The edges of the graph connect the instances represented as nodes. A well known graph theoretic algorithm [534] is based on the minimal spanning tree (MST) [248, 422].

2.4.3 Density-Based Methods

Most of the clustering methods, presented in the literature, group the objects based on the distances between them. These kind of methods can only find spherical clusters and unable to find the groups of non-spherical arbitrary shapes. Another method for clustering has been proposed based on density [75, 112], where the growth of the clusters continue till the density among the neighbouring points goes below some threshold, that is, for every sample in a cluster, atleast a specific number of points must be present around a given radius. This method is very useful to find out clusters of arbitrary shapes, and also to filter the noise and outliers. DBSCAN [123] and its extensions, OPTICS [16] are typical density based methods that grow clusters according to a density-based connectivity analysis. DENCLUE [189] is a method that clusters objects based on the H-analysis of the value distributions of density functions.

2.4.4 Grid-Based Methods

Grid-based methods [112, 567] create grid structures by partitioning the object space into finite number of cells, on which the operations are performed. The densities for each cell are calculated and the cells are sorted according to the densities. In this way, cluster centers are found out. The method is advantageous because of its fast processing time, as it depends on the number of cells in each dimension. WaveCluster [470], STRING [520] and OptiGrid [190] are some well known grid-based methods. In WaveCluster, wavelet transformation is applied for clustering analysis and it uses both grid-based and density-based methods.

2.4.5 Model-Based Clustering Methods

In model-based method [133], a hypothetical model is used to find out the best fit between itself and the given data. The algorithm searches the clusters by generating some density function that can model the spectral distribution of the samples. The method is also able to find out the number of clusters automatically from a data set containing noise and outliers. Maximum likelihood estimation (MLE) [372] is a very commonly used criterion to estimate the parameter inside the probability model. COBWEB [127] is an incremental conceptual learning algorithm that generates the clusters in the form of classification tree. Self-organizing feature map (SOM) [255] uses the concept of neural network, which generates the clusters by mapping high dimensional data into lower dimension (2-D or 3-D) feature map, and so it is helpful for data visualization.

2.4.6 Soft Computing Based Clustering Methods

Soft computing [218, 222, 384] is a recently coined term describing the symbiotic use of many emerging computing disciplines. The major components of soft computing are the computing disciplines of fuzzy logic [450], probabilistic reasoning, neural networks [64], and genetic algorithms [155]. Soft computing brings these disciplines under a common umbrella. It is a partnership in which each of the partners contributes a distinct methodology for addressing problems in its domain. In this perspective, the principal constituent methodologies in soft computing are complementary rather than competitive.

Soft computing combines the versatility of fuzzy logic to represent qualitative knowledge, with the data-driven efficiency of neural networks to provide fine-tuned adjustments via local search, with the ability of genetic algorithms to perform efficient coarse-granule global search. Leveraging the best features of these algorithms results in the development of hybrid algorithms that are superior to each of their underlying components.

2.4.6.1 Fuzzy C-Means

The conventional or hard clustering approaches generate different partitions, where each object belongs to one and only one cluster. Hence, the clusters produced by the hard clustering algorithms are disjoint. Fuzzy set theory was initially applied to clustering by Ruspini [450]. Fuzzy clustering [48,358,546] extends the partitioning scheme by associating each pattern with every cluster using a membership function. Hence, in fuzzy clustering [48,113], each point does not pertain to a given cluster, but has a degree of belongingness to a certain cluster, as in fuzzy logic. The output generated by such algorithms is considered as clustering, but not a partition. For each point x, we have a coefficient giving the degree of being in the kth cluster $u_k(x)$. Usually, the sum of those coefficients has to be 1.

Let, $X = \{x_1, \cdots, x_j, \cdots, x_n\}$ be the set of n objects and $V = \{v_1, \cdots, v_i, \cdots, v_c\}$ be the set of c centroids (means), where $x_j \in \Re^m$, $v_i \in \Re^m$, and $v_i \in X$. The fuzzy c-means (FCM) [48] provides a fuzzification of the hard c-means (HCM) [48].

FCM partitions X into c clusters by minimizing the objective function

$$J_F = \sum_{j=1}^{n}\sum_{i=1}^{c}(\mu_{ij})^{\acute{m}_1}||x_j - v_i||^2; \tag{2.36}$$

where $1 \leq \acute{m}_1 < \infty$ is the fuzzifier, v_i is the ith centroid corresponding to cluster β_i, $\mu_{ij} \in [0,1]$ is the probabilistic membership of the pattern x_j to cluster β_i, and $||.||$ is the distance norm.

Solving (2.36) with respect to μ_{ij}, we get

$$\mu_{ij} = \left[\sum_{k=1}^{c}\left(\frac{d_{ij}}{d_{kj}}\right)^{\frac{2}{\acute{m}_1-1}}\right]^{-1}; \text{ where } d_{ij}^2 = ||x_j - v_i||^2 \tag{2.37}$$

$$\text{subject to } \sum_{i=1}^{c}\mu_{ij} = 1, \forall j, \text{ and } 0 < \sum_{j=1}^{n}\mu_{ij} < n, \forall i.$$

The new centroid is calculated based on the fuzzy memberships of the objects. The modified centroid calculation for FCM is obtained by solving (2.36) with respect to v_i:

$$v_i^F = \frac{1}{n_i}\sum_{j=1}^{n}(\mu_{ij})^{\acute{m}_1}x_j; \text{ where } n_i = \sum_{j=1}^{n}(\mu_{ij})^{\acute{m}_1}. \tag{2.38}$$

The process begins by randomly choosing c objects as the centroids of the c clusters. The memberships are calculated based on the relative distance of the object x_j to the centroids $\{v_i\}$ by (2.38). After computing memberships of all the objects, the new centroids

of the clusters are calculated as per (2.38). The process stops when the centroids stabilize. That is, the centroids from the previous iteration are identical to those generated in the current iteration.

2.4.6.2 Possibilistic C-Means

In FCM, the memberships of an object are inversely related to the relative distance of the object to the cluster centroids. In effect, it is very sensitive to noise and outliers. Also, from the standpoint of "compatibility with the centroid", the membership of an object x_j in a cluster β_i should be determined solely by how close it is to the mean (centroid) v_i of the class, and should not be coupled with its similarity with respect to other classes. To alleviate this problem, Krishnapuram and Keller [261] introduced the possibilistic c-means (PCM). The membership matrix ν generated by the PCM is not a partition matrix in the sense that it does not satisfy the constraint

$$\sum_{i=1}^{c} \nu_{ij} = 1. \tag{2.39}$$

2.4.6.3 Rough-Fuzzy C-Means

Incorporating both fuzzy and rough sets, a hybrid method is proposed in [321], termed as rough-fuzzy c-means (RFCM). The RFCM adds the concept of fuzzy membership of fuzzy sets, and lower and upper approximations of rough sets into c-means algorithm. While the membership of fuzzy sets enables efficient handling of overlapping partitions, the rough sets deal with uncertainty, vagueness, and incompleteness in class definition.

Let, $\underline{A}(\beta_i)$ and $\overline{A}(\beta_i)$ be the lower and upper approximations of cluster β_i, and $B(\beta_i)(= \overline{A}(\beta_i) - \underline{A}(\beta_i))$ denotes the boundary region of cluster β_i. The RFCM algorithm partitions a set of n objects into c clusters by minimizing the objective function

$$J_{\mathrm{RF}} = w \times \mathcal{A}_1 + \tilde{w} \times \mathcal{B}_1 \tag{2.40}$$

$$\text{where } \mathcal{A}_1 = \sum_{i=1}^{c} \sum_{x_j \in \underline{A}(\beta_i)} ||x_j - v_i||^2; \quad \text{and } \mathcal{B}_1 = \sum_{i=1}^{c} \sum_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}_1} ||x_j - v_i||^2.$$

The parameters w and $\tilde{w}$ correspond to the relative importance of lower approximation and boundary region. Note that, μ_{ij} has the same meaning of membership as that in FCM.

2.4.6.4 Evolutionary Approaches for Clustering

In this type of algorithms, candidate solutions to the clustering problem are encoded as chromosomes. A solution is a valid k-partition of the data. The most commonly used evolutionary operators [71] are: selection, recombination, and mutation. Each transforms one or more input chromosomes into one or more output chromosomes. A chromosome's likelihood of surviving into the next generation depends on a fitness function, which is evaluated on a chromosome. The best-known evolutionary techniques are genetic algorithms (GA) [155,194], evolution strategies (ES) [462], and evolutionary programming (EP) [129]. Out of these three approaches, GAs have been most frequently used in clustering. GAs represent points in the search space as binary strings and uses the crossover operator to explore the search space. Mutation is used in GAs for the sake of completeness, that is, to make sure that no part of the search space is left unexplored. ESs and EPs differ from the GAs in solution representation and type of the mutation operator used; EP does not use a recombination operator, but only selection and mutation. Each of these three approaches has been used to solve the clustering problem by viewing it as a minimization of the squared error criterion.

2.5 Applications of Fuzzy Sets and Rough Sets

Fuzzy and rough sets have been successfully applied in feature selection [13, 83, 246, 323, 396, 501, 547, 576, 578], pattern classification [42, 201, 234, 266, 268, 361, 364, 458], clustering [48,51,72,113,238,358,380,398,450,453,454,505,546], image processing [29,137,212,224,247, 265, 318, 365, 387–390, 496, 537, 537], social media analysis [167], image segmentation [109], bioinformatics [30,79,84,216,289,348,349,355,357,529], industrial manufacturing [196,381], medicine [502], etc.

2.5.1 Fuzzy Sets in Pattern Recognition

Fuzzy sets have various applications in the domain of pattern recognition. Few of them are discussed below.

2.5.1.1 Feature Selection

Fuzzy sets have been successfully applied in various areas of feature selection. Papari et al. [396] developed a new two-stage approach for accurate modeling and prediction of tidal current. A novel fuzzy feature selection has been used in the proposed method to extract the most preferable features from the tidal current speed and direction data set. The selected features were further used to train a support vector regression for accurate

prediction. The setting parameters of the proposed model were trained by a new optimization algorithm based on the harmony search algorithm to get the most optimal training targets. The selection of nonredundant and relevant features of real-valued data sets is a highly challenging problem. A novel feature selection method has been presented in [323] based on fuzzy-rough sets by maximizing the relevance and minimizing the redundancy of the selected features. By introducing the fuzzy equivalence partition matrix, a novel representation of Shannon's entropy for fuzzy approximation spaces has been proposed to measure the relevance and redundancy of features suitable for real-valued data sets. The fuzzy equivalence partition matrix also offers an efficient way to calculate many more information measures, termed as f-information measures. Feature subset selection for classification is a well-known pattern recognition problem, where the reduction is expected to improve the performance of classification algorithms in terms of speed, accuracy and simplicity. The paper [501] focused on a problem called optimal fuzzy-valued feature subset selection (OFFSS), in which the quality-measure of a subset of features was defined by both the overall overlapping degree between two classes of examples and the size of feature subset.

2.5.1.2 Clustering

Fuzzy set theory was initially applied to clustering by Ruspini [450]. Here a general formulation of data reduction and clustering processes has been proposed. These procedures were regarded as mappings or transformations of the original space onto a representation or code space subjected to some constraints. Current clustering methods, as well as three other data reduction techniques, have been specified within the framework of this formulation. Fuzzy clustering [48, 358, 546] extends the partitioning scheme by associating each pattern with every cluster using a membership function. In [113], two fuzzy versions of the k-means, optimal and least squared error partitioning problem, have been formulated by Dunn for finite subsets X of a general inner product space. In both cases, the extremizing solutions have been shown to be fixed points of a certain operator T on the class of fuzzy, k-partitions of X, and simple iteration of T provided an algorithm which has the descent property relative to the least squared error criterion function. A class of fuzzy ISODATA clustering algorithms has been developed previously which includes fuzzy means. This class of algorithms has been generalized in [168] to include fuzzy covariances. The resulting algorithm closely resembles maximum likelihood estimation of mixture densities. Experimental results were presented which indicate that more accurate clustering may be obtained by using fuzzy covariances. In [239], Karayiannis presented a new approach to fuzzy clustering, which provides the basis for the development of the maximum entropy clustering algorithm (MECA). The derivation of the proposed algorithm was based on an objective function incorporating a measure of the entropy of the membership functions

and a measure of the distortion between the prototypes and the feature vectors. This formulation allows the gradual transition from a maximum uncertainty or minimum selectivity phase to a minimum uncertainty or maximum selectivity phase during the clustering process. The FCM algorithm and its extensions are usually affected by initializations and parameter selection with a number of clusters to be given a priori. Although there were some works to solve these problems in FCM, till the time when the algorithm was developed, there was no work for FCM to be simultaneously robust to initializations and parameter selection under free of the fuzziness index without a given number of clusters. In [546], a robust learning-based FCM framework has been constructed Yang and Nataliani, called as robust-learning FCM (RL-FCM) algorithm, which was free of the fuzziness index m and initializations without parameter selection, and was also able find the best number of clusters automatically.

2.5.1.3 Classification

Fuzzy sets have also been used for pattern classification [268]. Colorectal polyps affect a large percentage of the population all over the world, and they can be a basis for more serious conditions such as cancers. In [364], a fuzzy decision method for finding polyps on a colonoscopy image has been presented. As a first step, the image taken during the colonoscopy was cut into tiles of size N by N, thus a rough localization of the lesion within the picture can also be possible. The work [361] by Muniategui was on the development of a monitoring system of a spot welding production line. Here the process information and photographs of more than 150,000 parts have been used to improve the predictions of the previously developed fuzzy algorithm to predict the degradation state of the electrode. An alternative method has also been presented based on deep-learning that aimed at substituting the image analysis software to extract values associated with the quality level of the welded parts from photographs. In [458], a new automatic approach has been introduced, named as FEmoRec, for emotional context recognition from online social networks that applied a semantic similarity measure based on multi-layer perceptron neural net model. Here, it was assumed that a tweet may belong to many emotional categories with different membership degrees. The tweet has been classified by computing an emotion vector that represents the tweet's fuzzy membership values to Ekman's emotion classes. Intrusion detection system (IDS) is an auditing mechanism that analyzes the traffic system and applications to identify normal use of the system and an intrusion attempt and also it prevent security managers. In [42], a new genetic fuzzy system (GFS) model has been used, called as genetic programming fuzzy inference system for classification (GPFIS-Class), which was based on multi-gene genetic programming (MGGP). An efficient feature selection method has also been used to eliminate data redundancy and irrelevant features in order to analyze the huge data namely the NSL-KDD data set.

2.5.1.4 Image Analysis

Fuzzy sets have been successfully applied in different areas of image processing [29, 137, 247, 318, 387–390, 496, 537]. In [29], a new image retrieval scheme has been presented using visually significant point features. The clusters of points around significant curvature regions (high, medium, and weak type) have been extracted using a fuzzy set theoretic approach. Some invariant color features have been computed from these points to evaluate the similarity between images. A set of relevant and non-redundant features has been selected using the mutual information based minimum redundancy-maximum relevance framework. The relative importance of each feature has been evaluated using a fuzzy entropy based measure, which was computed from the sets of retrieved images marked relevant and irrelevant by the users. The paper [137] proposed an alternative and iterative image retrieval system which takes into account the subjectivity of human perception of visual content. The proposed system uses a dynamic similarity measure based on the Choquet integral. Both positive and negative user's feedback have been modeled by fuzzy sets, and were used to refine the feature relevance weights. The experimental results on more than 3000 texture images illustrate the learning behaviour of the retrieval system. A robust thresholding technique has been proposed in [318] for segmentation of brain MR images. It was based on the fuzzy thresholding techniques. Its aim was to threshold the gray level histogram of brain MR images by splitting the image histogram into multiple crisp subsets. The histogram of the given image has been thresholded according to the similarity between gray levels. The similarity has been assessed through a second order fuzzy measure such as fuzzy correlation, fuzzy entropy, and index of fuzziness. In [387], some new geometrical properties, e.g., length, breadth and index of area coverage (IOAC) of a fuzzy set along with their computational aspects have been introduced. An algorithm for providing both fuzzy and nonfuzzy segmentation based on these measures has also been proposed. A concept of correlation between two properties (fuzzy representations) of an image has been introduced in [388]. A set of algorithms for image segmentation (both fuzzy and nonfuzzy) has been formulated. The spatial information was taken care of by the following measures: transitional correlation and within-class correlation. Effectiveness of various fuzzy thresholding techniques (based on entropy of fuzzy sets, fuzzy geometrical properties, and fuzzy correlation) has been demonstrated in [389] on remotely sensed (IRS and SPOT) images. A new quantitative index for image segmentation using the concept of homogeneity within regions has been defined. Results were compared with those of probabilistic thresholding, and fuzzy c-means and hard c-means clustering algorithms, both in terms of index value (quantitatively) and structural details (qualitatively). Algorithms for automatic thresholding of grey levels (without reference to histogram) have been described in [390] using the terms index of fuzziness and entropy of a fuzzy set. Their values were

seen to be minimum when the crossover point of an S-function corresponds to boundary levels among different regions in image space. Methods for histogram thresholding based on the minimization of a threshold-dependent criterion function might not work well for images having multimodal histograms. In [496], an approach has been proposed to threshold the histogram according to the similarity between gray levels. Such a similarity has been assessed through a fuzzy measure.

2.5.2 Rough Sets in Pattern Recognition

There are various applications of rough sets available in the literature, which are in the domain of pattern recognition. Few of them are discussed below.

2.5.2.1 Feature Selection

Rough set attribute selection is utilized when the attributes are of discrete values [83, 327], whilst fuzzy-rough approaches are employed where attributes have continuous values. Methods in both categories use the concept of dependency for finding reducts. One of the well-known methods is the so called Quick-Reduct [83] that uses dependency. It does not guarantee to find minimal reduct as it employs a greedy algorithm which is a forward search and capable of being trapped in a local optimum. The rough set based MRMS criterion is also used in [327] for attribute selection task. Also, neighborhood rough sets [198] are found to be suitable for both numerical and categorical data sets. Neighborhood rough set based feature selection algorithm of Hu et al. [198] is based on Max-Dependency criterion. In [326], Maji and Paul used rough sets for the selection of molecular descriptors to predict biological activity of molecules in quantitative structure activity relationship (QSAR) data sets. The algorithm selects a set of effective molecular descriptors from a given QSAR dataset by maximizing both relevance and significance of the descriptors. One of the major issues of rough-fuzzy clustering is how to decide whether some objects are to be put in lower approximation or boundary region of a cluster. Rough hypercuboid approach [312,526] can implicitly partition the objects into lower approximation and boundary region of clusters, without the need of any user specified parameter.

Many different approaches have been employed to combine with a rough set feature selection in order to find the minimal reduct accurately in a reasonable time. These past researches can be categorized into two classes, namely, search-based strategies and evaluation measure based methods. Among the search-based strategies, Wang et al. [517] developed a rough set based method using nearest neighbour search for reducing computation complexity of indiscernibility relation in rough set attribute reduction. Ke et al. [246] proposed a modification on solution construction and pheromone update of ACO for finding minimal reduct. In [545], Yang and Yang proposed a method based on rough sets

and tree structure for finding reduct to save more space in feature selection. Ming [351] combined rough set and ACO for feature selection, while Yao [547] introduced a SVM based feature selection for credit scoring. Anaraki and Eftekhari [13] proposed a new stopping criterion for Quick-Reduct algorithm. Some evaluation measures based dimensionality reduction methods are also proposed using rough sets. Zhong and Dong [576] proposed a method based on rough sets and heuristic to evaluate features. In [578], Zhong and Skowron hybridized the generalized distribution table and rough sets for discovering classification rules. Chouchoulas and Shen [83] applied rough set theory for information filtering and information retrieval. Swiniarski and Skowron [487] merged rough set theory with principal component analysis for feature selection. In [564], Zhang et al. proposed one method by employing inclusion degree, evidence reasoning theory and rough sets for feature selection. Zhang and Yao [565] proposed a modification of lower approximation and introduced a feature support measure and reduct quality measure. Qian et al. [427] proposed a technique called "positive approximation" for compact feature selection computation. Parthalain et al. [400] used the tolerance rough set model to deal with real-valued data and proposed a distance measure to discover information from boundary region. Hu et al. [195] proposed a new attribute reduction based on discernibility matrix and attribute frequency information.

Combining fuzzy [150, 436, 438] and rough sets provides an important direction in reasoning with uncertainty for real valued data sets [110, 220]. They are complementary in some aspects. The generalized theories of rough-fuzzy computing have been applied successfully to feature selection of real valued data [199, 220, 323, 325]. The fuzzy-rough quick reduct algorithm of Jensen and Shen [220] is based on Max-Dependency criterion, while Maji and Pal [323] used the mRMR criterion for the feature selection based on f-information measures in fuzzy approximation spaces. In [312], a rough hypercuboid approach is presented for feature selection in approximation spaces. In [315], a rough-fuzzy MRMS based feature selection method was proposed, which was further improved in [316] using IT2 rough-fuzzy based approach. A rough-fuzzy based simultaneous attribute selection and feature extraction method was proposed in [314].

2.5.2.2 Clustering

The theory of rough sets has been widely used for clustering in approximation spaces [320, 321, 330]. In [289], Lingras and West introduced a new clustering method called rough c-means (RCM), which describes a cluster by a prototype (center) and a pair of lower and upper approximations. The lower and upper approximations have different weight parameters that are used to compute the new centers.

Combination of fuzzy sets and rough sets provides an effective way to solve this uncertainty problems [110, 220, 571]. Combining both rough and fuzzy sets, Mitra et al. [356]

proposed a new c-means algorithm, where each cluster consists of a fuzzy lower approximation and a fuzzy boundary. Each object in lower approximation takes a distinct weight, which is its fuzzy membership value. Integrating both rough and fuzzy sets, Maji and Pal proposed rough-fuzzy c-means (RFCM) algorithm [321], encapsulating two related and complementary, but distinct concepts used to represent the uncertainty in knowledge, vagueness (used in fuzzy set), and indiscernibility (used in rough sets). Each cluster in the RFCM consists of two disjoint regions, a crisp lower approximation, surrounded by an uncertain fuzzy boundary. Each object in lower approximation solely belongs to that cluster and has same weight, resulting a similar influence on the corresponding cluster and centroid updation. The algorithm provides the boundary region with gradual membership values, which can group the objects well in a data set with the presence of uncertainty. A cluster can be represented by its center, lower approximation and boundary region. A generalized hybrid algorithm, termed as rough-fuzzy possibilistic c-means (RFPCM), has also been proposed in [321]. The algorithm has been further extended to robust rough-fuzzy c-means (rRFCM) [330], where possibilistic lower approximation is surrounded by a probabilistic boundary region for any cluster. The possibilistic lower approximation helps to discover clusters having various shapes. In [408], a city block distance based rough-fuzzy clustering was proposed for identification of co-expressed microRNAs. In [145], a generalized interval type-2 fuzzy c-means algorithm was proposed based on rough hypercuboid approach.

2.5.2.3 Classification

The techniques of rough sets are also used for classification. In [266], an optimistic multi-granulation rough set based classification technique (OMGRS) has been developed for medical diagnosis. This research used rough set based data mining techniques for medical data to discover locally frequent diseases. Multi-granulation rough set provides efficient results than single granulation rough set model and soft rough set based classifier model. Chen et al. [79] developed a gene selection technique for tumor classification using neighborhood rough sets and entropy measures. The research addresses an entropy measure under the frame of neighborhood rough sets for tackling the uncertainty and noisy of gene expression data, resulting a discovery of compact gene subsets. In [274], a rough set based approach was introduced to discover classification rules through a process of knowledge induction, which selects decision rules with a minimal set of features for classification of real valued data. A rough set knowledge discovery framework was formulated for the analysis of interval-valued information systems converted from real-valued raw decision tables. In [499], the authors presented two classification approaches based on rough sets, that were able to learn decision rules from uncertain data. It was assumed that the uncertainty exists only in the decision attribute values of the decision table and is represented by the belief

functions. The first technique, named belief rough set classifier (BRSC), was based only on the basic concepts of the rough sets. The second, called belief rough set classifier, was more sophisticated. It was based on generalization distribution table (BRSC-GDT), which was a hybridization of the generalization distribution table and the rough sets (GDT-RS). The two classifiers aimed at simplifying the uncertain decision table (UDT) in order to generate significant decision rules for classification process.

2.5.2.4 Image Analysis

Rough set techniques can readily be applied for image analysis, including medical image processing and analysis. Kumar et al. [265] proposed a rough set based image-adaptive reference watermarking scheme based on DWT and SVD to solve the problem of image ambiguity and statically redundant wavelet coefficients. Here, the lower and upper approximation of wavelet sub-bands has been computed by considering the threshold wavelet coefficients. In [234], Jothi et al. developed a hybrid tolerance rough setfirefly based supervised feature selection for MRI brain tumor image classification. Different categories of features are extracted from the segmented MRI images, that is, shape, intensity and texture based features. Tolerance rough set (TRS) and firefly algorithm (FA) were used to select the imperative features of brain tumor. Garimella et al. [149] proposed few features based on information theory for image retrieval and a concept of probabilistic filtering for image processing. They also proposed a hybrid approach for image retrieval that combines annotation approach with content based image retrieval approach, and used rough set theory for audio/video object retrieval from multi-media databases. Banerjee and Maji [27] used rough sets for bias field correction in MR images using contraharmonic mean.

In [365] a soft fuzzy rough set-based MR brain image segmentation technique has been proposed that can handle the uncertainty related to the vagueness (fuzzy), indecernability (rough sets) in a parameterized representation (soft sets). To avoid local minima, it uses histogram based centroid initialization for choosing initial centroids for clustering. Islam et al. [212] developed a content-based image retrieval (CBIR) system and applied on some image databases using two MPEG-7 image descriptors. The method uses several sophisticated fuzzy-rough feature selection methods and combines the results of these methods to obtain a prominent feature subset for image representation for a particular query. Sarkar et al. [457] developed a rough possibilistic type-2 fuzzy c-means clustering for MR brain image segmentation. In [28], Banerjee and Maji presented a rough-probabilistic clustering and hidden Markov random field model for segmentation of HEp-2 cell and brain MR images. The algorithm integrates the merits of rough sets and a new probability distribution, called stomped normal (SN) distribution. Maji and Roy [334] proposed an algorithm for segmentation of brain tumor from MR images, which integrates the merits of rough-fuzzy computing and multi-resolution image analysis technique. To extract the scale-space

feature vector for each pixel of brain region, the dyadic wavelet analysis was used, while an unsupervised feature selection method, based on maximum relevance-maximum significance criterion, was used to select relevant and significant textural features for brain tumor segmentation. In [332], the authors introduced a rough-fuzzy clustering and multiresolution image analysis technique for text-graphics segmentation, where M-band wavelet packet was used to extract scale-space features for document image. In [333], a rough-fuzzy clustering and unsupervised feature selection technique was proposed for wavelet based MR image segmentation.

2.5.2.5 Bioinformatics

Rough sets have been applied [357] mainly to microarray gene expression data, in mining tasks like classification [348, 349], clustering [216, 289, 355], and feature selection [30, 84]. Classification rules (in if-then form) have been extracted from microarray data [349], using rough sets with supervised learning. The underlying assumption is that the associated genes are organized in an ontology, involving super and subclasses. This biological knowledge is utilized while generating rules in terms of the minimal characteristic features (reducts) of temporal gene expression profiles. A rule is said to cover a gene if the gene satisfies the conditional part, expressed as a conjunction of attribute-value pairs. The rules do not discriminate between the super and subclasses of the ontology, while retaining as much detail about the predictions without losing precision. Gastric tumor classification in microarray data was made using rough set based learning [348], implemented with ROSETTA involving GAs and dynamic reducts [529]. The fitness function incorporates measures involving the classification performance (discernibility) along with the size of the reduct. Thereby precedence provides to solutions having less number of attributes.

The rough c-means clustering algorithm can be used to the microarray gene expression data, where the concept of hard c-means is extended by viewing each cluster as an interval or rough set [289]. An evolutionary rough c-means clustering algorithm has been applied to microarray gene expression data [355]. Here, rough sets were used to model the clusters in terms of upper and lower approximations, while GAs were used to tune the threshold, and relative importance of upper and lower approximation parameters of the sets. The Davies-Bouldin clustering validity index [216] is used as the fitness function of the GA, which is minimized while arriving at an optimal partitioning. Expression profiling of miRNAs generates a huge volume of data. Complicated networks of miRNA-mRNA interaction create a big challenge for scientists to decipher this huge expression data. In order to extract meaningful information from expression data, the application of robust rough-fuzzy c-means (rRFCM) algorithm is presented in [406] to discover co-expressed miRNA clusters. The paper [409] presented a clustering algorithm in order to extract meaningful information from miRNA expression data, which judiciously integrated the merits of rough

sets, fuzzy sets, the c-means algorithm, and the normalized range-normalized city block distance to discover co-expressed miRNA clusters. The city block distance has been used to compute the membership functions of fuzzy sets and to find initial partition of a data set, and therefore helped to handle minute differences between two miRNA expression profiles.

An evolutionary rough feature selection algorithm has been developed in [30] for classifying microarray gene expression patterns. Since the data typically consists of a large number of redundant features, an initial redundancy reduction of the attributes was done to enable faster convergence. Thereafter rough sets theory was employed to generate reducts, which represent the minimal sets of non-redundant features that are capable of discerning between all objects, in a multiobjective framework. The effectiveness of the algorithm was demonstrated on three cancer data sets, viz., colon, lymphoma, and leukemia. While Chu et al. [84] generated a five-genes set for 100% correct classification on the lymphoma data in the NF framework, Banerjee et al. [30] obtained a misclassification for just two samples from the test data using a two-genes set. In [407], a μHEM based identification technique was proposed for differentially expressed miRNAs using hypercuboid equivalence partition matrix. Among the large number of miRNAs present in a microarray data, a modest number might be sufficient to classify human cancers. Hence, the identification of differentially expressed miRNAs is an important problem particularly for the data sets with large number of miRNAs and small number of samples. In [407], an approach has been reported for in silico identification of differentially expressed miRNAs from microarray expression data sets. Paul and Maji [410] presented a gene ontology based quantitative index, termed as degree of functional diversity (DoFD), to quantify the functional diversity of a set of genes selected by any gene selection algorithm. A new gene selection algorithm was also presented there, integrating the merits of both DoFD and RSMRMS, to select relevant and significant genes those are also functionally diverse. Another technique proposed by Paul and Maji [411] was regarding the use of gene expression and proteinprotein interaction data for identification of colon cancer related genes using f-information measures.

2.5.2.6 Data and Web Mining

Rough sets theory has been used for data and web mining using the feature selection and clustering tasks such as calculating centroids, fixing negative and positive regions, finding cluster labels, finding topics using the similarity measure calculation and cluster content discovery. The detection of outliers is a major issue in the clustering process where there is uncertainty to be addressed, such as whether the outlier can be put in an existing cluster or two outliers can form a new cluster. Document clustering is one of the important jobs in web mining, which can be done efficiently using rough sets.

Questier et al. [429] proposed one method where rough sets theory was used to con-

struct, under supervision, reducts for reducing the number of features without labels. Apart from the hierarchical method, the feature selection method can also be applied, combined with the clustering method, which is useful for document clustering. A heuristic algorithm based on rough set theory to learn about the feature subset has been proposed by Zhang et al. [287]. Starzyk et al. [213] introduced an algorithm to find the set of all reducts in a much shorter time, compared to the elimination method. An unsupervised algorithm was proposed by Jaganathan et al. [215] for clustering datasets without knowing the decision attribute for feature selection. In [2], rules were extracted to carry out two-fold data clustering, data discretization, and attribute selection. In [580], the authors analyzed a feature selection method with certain basic properties covering generalized rough sets, and a set of axioms proposed to characterize the covering lower approximation operation for document clustering. Rough set based feature selection has been implemented in [400], using a distance measure for document clustering. Rough set-based techniques were used by Mazlack et al. [340] to select clustering attributes. The ITDR was proposed by In-Kyoo Park et al. [397] as an alternative method for categorical data clustering, varying from the baseline method and rough set attribute reliance upon the rough entropy being calculated by the categorically-valued information system.

2.6 Conclusion

Due to the increment of data acquisition speed and decrement of the cost of both storage and computing power, people are now having access to large amount of data, that are mostly high-dimensional. In order to extract meaningful information, different pattern recognition and data mining techniques can be applied to the data set. Most of the data gathered from real life sources are imprecise in nature. In this background, fuzzy-rough sets is a well known tool that can handle uncertainties, vagueness, and incompleteness associated with data. Hence, some novel techniques can be developed to reduce the dimension in the uncertain environment. As mentioned in Section 2.1.1.3, dimensionality reduction using attribute selection is one of the important problems encountered in pattern recognition, machine learning, and related domains. A feature selection method is presented in next chapter, which is based on fuzzy-rough sets.

Chapter 3

Fuzzy-Rough Sets for Relevant and Significant Attribute Selection

3.1 Introduction

Attribute or feature selection is a process of selecting a map by which a sample in an m-dimensional measurement space is transformed into an object in a d-dimensional feature space, where $d < m$. The main objective of this task is to retain the optimum salient characteristics necessary for the pattern recognition process and to reduce the dimensionality of the measurement space so that effective and easily computable algorithms can be devised for efficient classification [101, 112, 382].

The problem of attribute selection has two aspects, namely, formulation of a suitable criterion to evaluate the goodness of a feature set and searching the optimal set in terms of the criterion [169]. In general, those features are considered to have optimal saliencies for which interclass (respectively, intraclass) distances are maximized (respectively, minimized). The criterion of a good feature is that it should be unchanging with any other possible variation within a class, while emphasizing differences that are important in discriminating between patterns of different classes [92, 393].

The conventional feature selection is based on the minimal classification error, which usually requires the maximal statistical dependency of the sample categories or class labels on the data distribution in the reduced feature space. This scheme is called maximal dependency or Max-Dependency, in which, the task of feature selection is to find a feature subset from the whole feature set, which jointly have the largest dependency on the target class [34,202,269]. However, the main drawback of this approach is the slow computational speed. Also, the joint dependency of the features for high dimensional real life data sets cannot be estimated correctly [105, 416]. Hence, although Max-Dependency feature selection might be useful to select a very small number of features, it is not appropriate

for real life applications where the aim is to achieve high classification accuracy with a reasonably compact set of features.

As the Max-Dependency criterion is hard to implement, an alternative is to select features based on maximal relevance or Max-Relevance criterion. Max-Relevance is to search a set of features that approximates Max-Dependency criterion with the mean value of all dependency values between individual feature and target class label. However, Max-Relevance criterion does not consider the joint effect of features on the target class. Moreover, it is likely that features selected according to Max-Relevance could have rich redundancy, that is, the dependency among these features could be large [89, 173].

Some feature selection methods have been reported to reduce redundancy among the selected features directly based on minimal redundancy or Min-Redundancy criterion [61, 188] or indirectly based on maximal significance or Max-Significance criterion [326–328]. Min-Redundancy criterion has also been studied in principal component analysis (PCA) and independent component analysis (ICA) [112, 207], which aims to find nonredundant features in a transformed domain. Combining redundancy or significance criterion with relevance criterion, minimal redundancy-maximal relevance (mRMR) [105, 323, 416] and maximal relevance-maximal significance (MRMS) [326–328] criteria have been proposed to select relevant and nonredundant or significant features.

An optimal feature subset selected by a feature selection algorithm is always relative to a certain feature evaluation index. In general, different indices may lead to different optimal feature subsets [92, 93]. However, every index tries to measure the discriminating ability of a feature or a subset of features to distinguish different class labels or sample categories. To compute the effectiveness of a feature or a subset of features, different statistical measures, Euclidean distance [105], mutual information [34, 202, 269], class separability index [101], Davies-Bouldin (DB) index [94], Dunn index [50], and fuzzy feature evaluation index (FFEI) [385] are widely used. One of the main problems in real life data analysis is uncertainty. Some of the sources of this uncertainty include incompleteness and vagueness in class definitions. In this background, the possibility concept introduced by the theory of rough sets [414] has gained popularity in modeling and propagating uncertainty. It has been applied to reasoning with uncertainty, fuzzy rule extraction and modeling, classification, clustering, and feature selection [222, 311, 319–322, 324, 325, 392, 414].

The theory of rough sets can be used to find a subset of informative features from the original attributes of a given data set with discretized attribute values [83, 327]. While the quick reduct algorithm of Chouchoulas and Shen [83] is based on the principle of Max-Dependency criterion, the MRMS criterion is used in [327] for attribute selection task. However, there are usually real valued data and fuzzy information in real world applications. In the theory of rough sets, the real valued features are divided into several discrete partitions and the dependency or quality of approximation of a feature is calculated. The

inherent error that exists in discretization process is of major concern in the computation of the dependency of real valued features. Combining fuzzy [150, 436, 438] and rough sets provides an important direction in reasoning with uncertainty for real valued data sets [110, 220]. They are complementary in some aspects. The generalized theories of rough-fuzzy computing have been applied successfully to feature selection of real valued data [199, 220, 323, 325]. Also, neighborhood rough sets [198] are found to be suitable for both numerical and categorical data sets. The fuzzy-rough quick reduct algorithm of Jensen and Shen [220] and neighborhood rough set based feature selection algorithm of Hu et al. [198] are based on Max-Dependency criterion, while the feature selection method based on f-information measures in fuzzy approximation spaces of Maji and Pal [323] uses the mRMR criterion.

In this regard, a fuzzy-rough feature selection method (FR-MRMS) is presented in this chapter, integrating judiciously the merits of fuzzy-rough sets and MRMS criterion, to provide a means by which real valued noisy features can be effectively reduced without the need for user-specified information. The proposed method selects a subset of features or condition attributes from the whole feature set by maximizing the relevance and significance of the selected features. Both relevance and significance of the features are computed using the concept of fuzzy positive regions of fuzzy-rough sets. Hence, the only information required in the proposed feature selection method is in the form of fuzzy partitions or information granules for each condition attribute. The π function in the one dimensional form is used to generate fuzzy information granules corresponding to each condition attribute, where the centers and radii of the π functions can be determined automatically from the distribution of training patterns. The fuzzy positive regions of decision attributes or class labels are computed based on the concept of fuzzy equivalence partition matrix [323]. The method can be applied to regression as well as classification problems with continuous decision attributes. The effectiveness of the proposed FR-MRMS method, along with a comparison with other methods, is demonstrated on a set of benchmark and microarray gene expression data sets using the predictive accuracy of nearest neighbor rule, support vector machine, and decision tree. Some of the results, presented in this chapter, are also reported in [313, 315].

The structure of rest of this chapter is as follows: Section 3.2 presents the proposed fuzzy-rough attribute selection method. Classifiers and data sets used in the experiments are described in Section 3.3. A few case studies and a comparison with other methods are reported in Section 3.4. Concluding remarks are given in Section 3.5.

3.2 Fuzzy-Rough Attribute Selection Method

This section presents a feature selection algorithm, integrating judiciously the theory of fuzzy-rough sets and merits of the MRMS criterion.

3.2.1 Fuzzy-Rough MRMS Method

The real life high dimensional data set may contain a number of irrelevant and insignificant features. The presence of such features may lead to a reduction in the useful information. The selected feature subset should contain the features those have high relevance with the classes and high significance in the feature set. The features with high relevance are expected to be able to predict the classes of the samples. In contrast, the presence of insignificant features in the subset may degrade the prediction capability. On the other hand, a feature set with high relevance and high significance enhances the predictive capability. Accordingly, a measure is required that can assess the effectiveness of a feature set. In this chapter, the theory of fuzzy-rough sets is used to select relevant and significant features from a data set.

Let $\mathbb{C} = \{\mathcal{A}_1, \cdots, \mathcal{A}_i, \cdots, \mathcal{A}_j, \cdots, \mathcal{A}_m\}$ be the set of m condition attributes or features of a given data set and $\mathbb{S} \subseteq \mathbb{C}$ with cardinality $d < m$ is the set of selected features. Define $\gamma_{\mathcal{A}_i}(\mathbb{D})$ as the relevance of the feature $\mathcal{A}_i$ with respect to the class labels $\mathbb{D}$ while $\sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_i)$ as the significance of the feature $\mathcal{A}_i$ with respect to the set $\{\mathcal{A}_i, \mathcal{A}_j\}$. The average relevance of all selected features is, therefore, given by

$$\mathrm{R} = \frac{1}{|\mathbb{S}|} \sum_{\mathcal{A}_i \in \mathbb{S}} \gamma_{\mathcal{A}_i}(\mathbb{D}); \tag{3.1}$$

while the average significance among the selected features is

$$\mathrm{S} = \frac{\sum\limits_{\mathcal{A}_i \neq \mathcal{A}_j \in \mathbb{S}} \{\sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_i) + \sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_j)\}}{|\mathbb{S}|(|\mathbb{S}| - 1)} \tag{3.2}$$

$$\text{that is, } \mathrm{S} = \frac{\sum\limits_{\mathcal{A}_i \neq \mathcal{A}_j \in \mathbb{S}} 2\gamma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}) - \{\gamma_{\mathcal{A}_i}(\mathbb{D}) + \gamma_{\mathcal{A}_j}(\mathbb{D})\}}{|\mathbb{S}|(|\mathbb{S}| - 1)}. \tag{3.3}$$

Therefore, the problem of selecting a set $\mathbb{S}$ of d relevant and significant features from the whole set $\mathbb{C}$ of m features is equivalent to optimize R and S simultaneously:

$$\max \mathcal{C}(\mathrm{R}, \mathrm{S}), \qquad \mathcal{C} = \omega \mathrm{R} + (1 - \omega)\mathrm{S}. \tag{3.4}$$

where $0 \leq \omega \leq 1$ is a weight parameter and the operator $\mathcal{C}(\mathrm{R}, \mathrm{S})$ is defined to combine R and S.

3.2.2 Computation of Relevance and Significance

Both relevance and significance of a feature are calculated based on fuzzy-rough set theory. Given a finite set $\mathbb{U}$, $\mathbb{C}$ is a fuzzy attribute set in $\mathbb{U}$, which generates a fuzzy equivalence partition on $\mathbb{U}$. If c denotes the number of fuzzy equivalence classes generated by the fuzzy equivalence relation and n is the number of objects in $\mathbb{U}$, then c-partitions of $\mathbb{U}$ can be arrayed as a $(c \times n)$ matrix $\mathbb{M}_{\mathbb{C}}$, termed as fuzzy equivalence partition matrix (FEPM) [323], which is denoted by

$$\mathbb{M}_{\mathbb{C}} = \begin{pmatrix} m_{11}^{\mathbb{C}} & m_{12}^{\mathbb{C}} & \dots & m_{1n}^{\mathbb{C}} \\ m_{21}^{\mathbb{C}} & m_{22}^{\mathbb{C}} & \dots & m_{2n}^{\mathbb{C}} \\ \dots & \dots & \dots & \dots \\ m_{c1}^{\mathbb{C}} & m_{c2}^{\mathbb{C}} & \dots & m_{cn}^{\mathbb{C}} \end{pmatrix} \tag{3.5}$$

where $m_{ij}^{\mathbb{C}} \in [0, 1]$ represents the membership of object x_j in the ith fuzzy equivalence partition or class F_i.

Definition 3.1 *The relevance of the condition attribute $\mathcal{A}_i$ with respect to the decision attribute set $\mathbb{D}$ can be defined as follows:*

$$\gamma_{\mathcal{A}_i}(\mathbb{D}) = \frac{1}{n}\sum_{j=1}^{n} \kappa_j; \qquad 0 \leq \gamma_{\mathcal{A}_i}(\mathbb{D}) \leq 1; \qquad \text{and}$$
$$\kappa_j = \sup_k\{\sup_s\{\min\{m_{sj}^{\mathcal{A}_i}, \inf_l\{\max\{1 - m_{sl}^{\mathcal{A}_i}, m_{kl}^{\mathbb{D}}\}\}\}\}\}. \tag{3.6}$$

The family of normal fuzzy sets produced by a fuzzy partitioning of the universe of discourse can play the role of fuzzy equivalence classes. In general, the π function in the one dimensional form is used to assign membership values to different fuzzy equivalence classes for the input features. A fuzzy set with membership function $\pi(x; \bar{c}, \sigma)$ represents a set of points clustered around $\bar{c}$, where

$$\pi(x; \bar{c}, \sigma) = \begin{cases} 2\left(1 - \frac{||x-\bar{c}||}{\sigma}\right)^2 & \text{for } \frac{\sigma}{2} \leq ||x - \bar{c}|| \leq \sigma \\ 1 - 2\left(\frac{||x-\bar{c}||}{\sigma}\right)^2 & \text{for } 0 \leq ||x - \bar{c}|| \leq \frac{\sigma}{2} \\ 0 & \text{otherwise} \end{cases} \tag{3.7}$$

where $\sigma > 0$ is the radius of the π function with $\bar{c}$ as the central point and $||\cdot||$ denotes the Euclidean norm. When the pattern x lies at the central point $\bar{c}$ of a class, then $||x - \bar{c}|| = 0$

and its membership value is maximum, that is, $\pi(\bar{c};\bar{c},\sigma) = 1$. The membership value of a point decreases as its distance from the central point $\bar{c}$, that is, $||x - \bar{c}||$ increases. When $||x - \bar{c}|| = \left(\frac{\sigma}{2}\right)$, the membership value of the object x is 0.5 and this is called a crossover point [395].

The $c \times n$ FEPM $\mathbb{M}_{\mathcal{A}_i}$, corresponding to the ith feature $\mathcal{A}_i$, can be calculated from the c-fuzzy equivalence classes of the objects $x = \{x_1, \ldots, x_j, \ldots, x_n\}$, where

$$m_{kj}^{\mathcal{A}_i} = \frac{\pi(x_j;\bar{c}_k,\sigma_k)}{\sum_{l=1}^{c}\pi(x_j;\bar{c}_l,\sigma_l)}. \tag{3.8}$$

In effect, each position $m_{kj}^{\mathcal{A}_i}$ of the FEPM $\mathbb{M}_{\mathcal{A}_i}$ must satisfy the following conditions:

$$m_{kj}^{\mathcal{A}_i} \in [0,1];\ \sum_{k=1}^{c} m_{kj}^{\mathcal{A}_i} = 1, \forall j \text{ and}$$

$$\text{for any value of } k, \text{ if } s = \arg\max_j\{m_{kj}^{\mathcal{A}_i}\},\ \text{then } \max_j\{m_{kj}^{\mathcal{A}_i}\} = \max_l\{m_{ls}^{\mathcal{A}_i}\} > 0.$$

In the present work, three fuzzy equivalence classes, namely, low, medium, and high, are considered. Corresponding to three fuzzy sets ($c = 3$), the following relations hold:

$$\bar{c}_1 = \bar{c}_{\text{low}}(\mathcal{A}_i);\ \bar{c}_2 = \bar{c}_{\text{medium}}(\mathcal{A}_i);\ \bar{c}_3 = \bar{c}_{\text{high}}(\mathcal{A}_i)$$
$$\sigma_1 = \sigma_{\text{low}}(\mathcal{A}_i);\ \sigma_2 = \sigma_{\text{medium}}(\mathcal{A}_i);\ \sigma_3 = \sigma_{\text{high}}(\mathcal{A}_i).$$

Each real valued feature in quantitative form can be assigned to different fuzzy equivalence classes in terms of membership values using the π fuzzy set with appropriate $\bar{c}$ and σ. The centers and radii of the π functions along each feature axis can be determined automatically from the distribution of training patterns or objects [394]. Let $\bar{m}_i$ be the mean of the objects $x = \{x_1, \ldots, x_j, \ldots, x_n\}$ along the ith feature $\mathcal{A}_i$. Then $\bar{m}_{i_l}$ and $\bar{m}_{i_h}$ are defined as the means, along the ith feature, of the objects having co-ordinate values in the range $[\mathcal{A}_{i_{\min}}, \bar{m}_i)$ and $(\bar{m}_i, \mathcal{A}_{i_{\max}}]$, respectively, where $\mathcal{A}_{i_{\max}}$ and $\mathcal{A}_{i_{\min}}$ denote the upper and lower bounds of the dynamic range of feature $\mathcal{A}_i$ for the training set. For three fuzzy sets low, medium, and high, the centers and corresponding radii are as follows [394]:

$$\bar{c}_{\text{low}}(\mathcal{A}_i) = \bar{m}_{i_l};\ \bar{c}_{\text{medium}}(\mathcal{A}_i) = \bar{m}_i;\ \bar{c}_{\text{high}}(\mathcal{A}_i) = \bar{m}_{i_h} \tag{3.9}$$

$$\begin{aligned} \sigma_{\text{low}}(\mathcal{A}_i) &= 2(\bar{c}_{\text{medium}}(\mathcal{A}_i) - \bar{c}_{\text{low}}(\mathcal{A}_i)) \\ \sigma_{\text{high}}(\mathcal{A}_i) &= 2(\bar{c}_{\text{high}}(\mathcal{A}_i) - \bar{c}_{\text{medium}}(\mathcal{A}_i)) \end{aligned} \tag{3.10}$$

$$\sigma_{\text{medium}}(\mathcal{A}_i) = \frac{\eta}{(\mathcal{A}_{i_{\max}} - \mathcal{A}_{i_{\min}})} [\sigma_{\text{low}}(\mathcal{A}_i)(\mathcal{A}_{i_{\max}} - c_{\text{medium}}(\mathcal{A}_i)) + \sigma_{\text{high}}(\mathcal{A}_i)(c_{\text{medium}}(\mathcal{A}_i) - \mathcal{A}_{i_{\min}})] \tag{3.11}$$

where η is a multiplicative parameter controlling the extent of the overlapping. The distribution of objects along each feature axis is taken into account, while computing the corresponding centers and radii of three fuzzy sets. Also, the amount of overlap between three fuzzy sets can be different along the different axis, depending on the distribution of the objects.

To calculate the significance of a condition attribute, the joint relevance $\gamma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D})$ between two attributes $\mathcal{A}_i$ and $\mathcal{A}_j$ needs to be computed. The construction of resultant FEPM $\mathbb{M}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ is necessary for computing the joint relevance. Let c_i and c_j be the number of fuzzy equivalence classes generated by the condition attributes $\mathcal{A}_i$ and $\mathcal{A}_j$, respectively. If r is the number of resultant fuzzy equivalence partitions, then the $(r \times n)$ FEPM $\mathbb{M}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ can be computed as follows:

$$\mathbb{M}_{\{\mathcal{A}_i,\mathcal{A}_j\}} = \mathbb{M}_{\mathcal{A}_i} \cap \mathbb{M}_{\mathcal{A}_j} \tag{3.12}$$

where $m_{kl}^{\{\mathcal{A}_i,\mathcal{A}_j\}} = m_{pl}^{\mathcal{A}_i} \cap m_{ql}^{\mathcal{A}_j}$, $k = (p-1)c_j + q$, and $\max\{c_i, c_j\} \leq r \leq c_i c_j$. In the present work, three fuzzy equivalence classes are considered, that is, $c_i = c_j = 3$.

3.2.3 FR-MRMS Algorithm

Following greedy algorithm is used to solve (3.4) based on the theory of fuzzy-rough sets:

- *Input:* Original set $\mathbb{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_k, \ldots, \mathcal{A}_m\}$.
- *Output:* Reduced set $\mathbb{S}$.

1. Initialize $\mathbb{C} \leftarrow \{\mathcal{A}_1, \ldots, \mathcal{A}_i, \ldots, \mathcal{A}_j, \ldots, \mathcal{A}_m\}, \mathbb{S} \leftarrow \emptyset$.
2. Calculate the centers and radii of three π fuzzy sets for each feature $\mathcal{A}_i \in \mathbb{C}$ according to (3.9), (3.10), and (3.11).
3. Construct the FEPM $\mathbb{M}_{\mathcal{A}_i}$ for each feature $\mathcal{A}_i \in \mathbb{C}$ according to (3.8).
4. Calculate the relevance $\gamma_{\mathcal{A}_i}(\mathbb{D})$ of each feature $\mathcal{A}_i \in \mathbb{C}$ according to (3.6).
5. Select feature $\mathcal{A}_i$ as most relevant feature that has highest relevance value $\gamma_{\mathcal{A}_i}(\mathbb{D})$. In effect, $\mathcal{A}_i \in \mathbb{S}$ and $\mathbb{C} = \mathbb{C} \setminus \mathcal{A}_i$.
6. Repeat the following four steps until $\mathbb{C} = \emptyset$ or the desired number of features d is selected.

(a) Construct resultant FEPM $\mathbb{M}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ for each remaining feature $\mathcal{A}_j \in \mathbb{C}$ and selected feature $\mathcal{A}_i \in \mathbb{S}$ using (3.12).

(b) Calculate the significance of $\mathcal{A}_j \in \mathbb{C}$ with respect to each of the selected features $\mathcal{A}_i \in \mathbb{S}$ as follows:

$$\sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_j) = \gamma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}) - \gamma_{\mathcal{A}_i}(\mathbb{D}). \tag{3.13}$$

(c) Remove $\mathcal{A}_j$ from $\mathbb{C}$ if $\sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_j) = 0$ for any feature $\mathcal{A}_i \in \mathbb{S}$.

(d) From the remaining features of $\mathbb{C}$, select feature $\mathcal{A}_j$ that maximizes the following condition:

$$\omega\gamma_{\mathcal{A}_j}(\mathbb{D}) + \frac{(1-\omega)}{|\mathbb{S}|} \sum_{\mathcal{A}_i \in \mathbb{S}} \sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_j). \tag{3.14}$$

As a result of that, $\mathcal{A}_j \in \mathbb{S}$ and $\mathbb{C} = \mathbb{C} \setminus \mathcal{A}_j$.

7. Stop.

3.2.4 Complexity of the Algorithm

The FR-MRMS has low computational complexity with respect to the number of features and samples in the original data set. The steps 2, 3, and 4 of the proposed algorithm are executed m times for m attributes. The complexity to compute the centers and radii of three fuzzy sets for each attribute, which is carried out in step 2, is $\mathcal{O}(n)$. The construction of the FEPM of each feature performed in step 3 has $\mathcal{O}(nc)$ time complexity. The computation of the relevance of each feature is carried out in step 4, which has $\mathcal{O}(nc\tilde{c})$ time complexity, where $\tilde{c}$ represents the number of fuzzy equivalence classes of decision attribute. Hence, the overall time complexity of steps 2, 3, and 4 for m features is $\mathcal{O}(mnc\tilde{c})$.

The selection of most relevant feature from the set of m features, which is carried out in step 5, has a complexity $\mathcal{O}(m)$. There is only one loop in step 6 of the proposed feature selection method, which is executed $(d-1)$ times, where d represents the number of selected features. The construction of the resultant FEPM, which is carried out in step 6.(a), and the computation of significance of a candidate feature with respect to an already-selected feature, which is carried out in step 6.(b), have $\mathcal{O}(nc^2)$ and $\mathcal{O}(nc^2\tilde{c})$ time complexity, respectively. If $\acute{m} < m$ represents the cardinality of the already-selected feature set, the total complexity of steps 6.(a) and 6.(b) is $\mathcal{O}((m-\acute{m})(nc^2\tilde{c}))$. The selection of a feature from $(m-\acute{m})$ candidate features by maximizing both relevance and significance, which is carried out in step 6.(d), has a complexity $\mathcal{O}(m-\acute{m})$. Hence, the total complexity to execute the loop $(d-1)$ times is $\mathcal{O}((d-1)(m-\acute{m})(nc^2\tilde{c}))$.

In effect, the selection of a set of d relevant and significant features from the whole set of m features using the proposed fuzzy-rough set based first order incremental search method has an overall computational complexity of $\mathcal{O}(mndc^2\tilde{c})$.

3.3 Classifiers and Data Sets Used

This section presents a brief descriptions of the classifiers and data sets used.

3.3.1 Classifiers

Three pattern classifiers, k-nearest neighbor (K-NN) rule [112], support vector machine (SVM) [508], and C4.5 decision tree [431] are used to evaluate the performance of different dimensionality reduction methods with respect to several real life data sets.

3.3.1.1 K-Nearest Neighbor Rule

The k-nearest neighbor (K-NN) rule [112] is used for evaluating the effectiveness of the reduced feature set for classification. It classifies samples based on closest training samples in the feature space. A sample is classified by a majority vote of its k-neighbors, with the sample being assigned to the class most common amongst its k-nearest neighbors. The value of k, chosen for the K-NN [112], is the square root of number of samples in training set.

3.3.1.2 Support Vector Machine

The support vector machine (SVM) [508] is a relatively new and promising classification method. It is a margin classifier that draws an optimal hyperplane in the feature vector space; this defines a boundary that maximizes the margin between data samples in different classes, therefore leading to good generalization properties. A key factor in the SVM is to use kernels to construct nonlinear decision boundary. In the present work, linear kernels are used.

3.3.1.3 C4.5 Decision Tree

The C4.5 [431] is a popular decision tree-based classification algorithm. It is used for evaluating the effectiveness of reduced feature set for classification. The selected feature set are fed to the C4.5 for building classification models. The C4.5 is used here because it performs feature selection in the process of training and the classification models it builds are represented in the form of decision trees, which can be further examined.

3.3.2 Description of Data Sets

This subsection reports some benchmark data sets that are used to evaluate the performance of different methods. While Satimage, Segmentation, Isolet, and Multiple Features data sets are downloaded from the *UCI Machine Learning Repository* [134], Breast Cancer I, Breast Cancer II, Colon Cancer, Lung Cancer, Leukemia I, and Leukemia II data sets are available at the *Kent Ridge Bio-medical Data Set Repository* [1].

3.3.2.1 Satimage

The database is a tiny sub-area of a scene, consisting of 82×100 pixels, each pixel covering an area on the ground of approximately 80×80 meters. The information given for each pixel consists of the class value and the intensities in four spectral bands, from the green, red, and infra-red regions of the spectrum. The data set contains 6435 examples: 4435 training and 2000 testing, with 36 real valued attributes and 6 classes.

3.3.2.2 Segmentation

This data set contains instances that are drawn randomly from a database of 7 outdoor images. The images are hand segmented to create a classification for every pixel, where each instance is a 3×3 region. The data set contains 3310 examples: 210 training and 2100 testing, with 18 continuous attributes and 7 classes.

3.3.2.3 Colon Cancer

The colon cancer data set contains expression levels of 2000 genes and 62 samples from two classes: 40 tumor and 22 normal colon tissues.

3.3.2.4 Breast Cancer I

This data set contains expression levels of 7129 genes in 49 breast tumor samples. The samples are classified according to their estrogen receptor (ER) status: 25 samples are ER positive while other 24 samples are ER negative.

3.3.2.5 Breast Cancer II

In this data set, relapse or non relapse of metastases in patients after initial diagnosis for interval of at least 5 years has been classified in breast cancer patients. Total 97 samples are given: 78 training and 19 testing, with 46 patients developed distance metastases within 5 years, labeled as relapse, while 51 remained healthy, labeled as non-relapse. The data set consists of 24481 genes [1].

3.3.2.6 Lung Cancer

This data set contains 181 tissue samples: among them 31 are malignant pleural mesothelioma and rest 150 adenocarcinoma of the lung. Each sample is described by the expression levels of 12533 genes.

3.3.2.7 Leukemia I

It is an Affymetrix high density oligonucleotide array that contains 7070 genes and 72 samples from two classes of leukemia: 47 acute lymphoblastic leukemia and 25 acute myeloid leukemia.

3.3.2.8 Leukemia II

This data set consists of gene expression profiles of 215 training and 112 testing samples classified into 7 classes, six subtypes of pediatric acute lymphoblastic leukemia and one that contains diagnostic samples that did not fit into any one of the six groups. The data set contains total 12558 genes.

3.3.2.9 Isolet

The data set consists of several spectral coefficients of utterances of English alphabets by 150 subjects. There are 617 real valued features with 7797 instances and 26 classes.

3.3.2.10 Multiple Features

Multiple features data set consists of features of handwritten numerals ('0'-'9') extracted from a collection of Dutch utility maps. 200 patterns per class (for a total of 2,000 patterns) have been digitized in binary images. Total 649 attributes are there in the data set.

To compute the classification accuracy of the K-NN, SVM, and C4.5, both training-testing and 10 fold cross-validation (CV) are performed. The 10-fold CV is performed on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features data sets, while the training-testing is done on Satimage, Segmentation, Leukemia II, and Breast Cancer II data sets.

3.4 Experimental Results and Discussion

The performance of proposed FR-MRMS algorithm is extensively studied and compared with that of different feature selection and extraction algorithms. The algorithms compared are InfoGain [430] and mutual information based mRMR framework (classical

mRMR) [416]; rough set based quick reduct [83] and MRMS framework [327]; fuzzy-rough set based quick reduct [220] and mRMR method (fuzzy-rough mRMR) [323]; margin based approaches such as relevance in estimating features (RELIEF) [252] and iterative search margin based algorithm (SIMBA) [152]; and existing feature extraction algorithms, namely, PCA, ICA, and linear discriminant analysis (LDA) [112]. The performance of fuzzy-rough sets and the MRMS criterion is also compared with that of other feature evaluation criteria, namely, Max-Relevance and Max-Dependency, several existing feature evaluation indices, namely, class separability index [101], DB index [94], Dunn index [50], and FFEI [385], which are briefly described in Appendix A, and various rough set models such as classical and neighborhood rough sets.

All the algorithms are implemented in C language and run in Ubuntu 11.04 environment with 64 bit support having machine configuration of Pentium Core 2 Quad 2.66 GHz with 4 MB L2 cache, and 4 GB DDR2 RAM. The value of multiplicative parameter η in (3.11) of fuzzy-rough sets is set to 1.5, while the weight parameter ω in (3.14) and that of mRMR method are set to 0.5. The discretization method reported in [310] is used to generate equivalence classes of Pawlak's or classical rough sets. For the data set with small number of features, 80% of total features is selected, while fifty top ranked features are considered for the data set with large number of features. In all cases, the result is presented for highest classification accuracy.

3.4.1 Statistical Significance Test

In case of 10-fold CV, the means and standard deviations of the classification accuracy of the K-NN, SVM, and C4.5 are computed for Breast I, Colon, Lung, Leukemia I, Isolet, and Multiple Features data sets. Tests of significance are performed for the inequality of means (of the classification accuracy of the SVM, K-NN, and C4.5) obtained using the fuzzy-rough MRMS method and other approaches. Since both mean pairs and variance pairs are unknown and different, a generalized version of t-test is used here. The above problem is the classical Behrens-Fisher problem in hypothesis testing. The test statistic, described and tabled in [20], is of the form

$$t = \frac{\mu_1 - \mu_2}{\sqrt{\lambda_1 {\sigma_1}^2 + \lambda_2 {\sigma_2}^2}} \tag{3.15}$$

where μ_1, μ_2 are the means, σ_1, σ_2 the standard deviations, and $\lambda_1 = 1/n_1$, $\lambda_2 = 1/n_2$, n_1, n_2 are number of observations. Tables 3.2 - 3.7 report the individual means and standard deviations, and the value of test statistic computed. The corresponding tabled value is 1.81 at an error probability level of 0.05. If the computed value is greater than the tabled value, the means are significantly different.

Table 3.1: Comparative Performance Analysis on Satimage, Segmentation, Leukemia II, and Breast II

	Different Criteria	Evaluation Indices	Satimage			Segmentation			Leukemia II			Breast II		
			K-NN	SVM	C4.5	K-NN	SVM	C4.5	K-NN	SVM	C4.5	K-NN	SVM	C4.5
Different Rough Sets	Max Relevance	Classical	69.2/8	67.3/8	67.5/8	63.2/11	57.9/13	63.2/8	81.3/12	81.3/13	81.3/14	67.9/16	78.9/13	77.5/15
		Neighborhood	77.7/13	74.7/10	79.6/17	78.9/13	73.7/11	73.7/12	83.0/25	80.4/11	80.4/21	76.8/17	78.9/14	77.5/11
		Fuzzy-Rough	77.9/10	76.4/9	80.0/11	78.9/13	78.9/13	78.9/14	83.0/16	82.1/14	82.1/15	82.1/22	80.9/19	78.8/12
	Max Dependency	Classical	70.4/8	67.3/9	67.5/8	63.2/10	57.9/6	68.4/8	82.1/15	82.1/12	82.1/12	67.9/12	82.5/14	81.8/13
		Neighborhood	83.2/11	81.7/10	81.5/21	79.0/13	73.7/12	73.7/9	85.7/21	83.0/12	83.0/14	81.5/16	82.5/11	82.8/17
		Fuzzy-Rough	83.2/12	83.0/11	82.8/15	78.9/13	78.9/13	78.9/13	85.7/17	84.8/13	84.8/13	84.2/12	83.8/15	82.8/11
	MRMS	Classical	74.0/9	73.9/10	74.1/10	72.7/9	74.1/13	74.7/12	84.8/15	85.7/15	85.7/15	84.2/11	84.2/14	84.2/15
		Neighborhood	83.4/10	**85.0/16**	**85.2/11**	80.5/14	82.6/15	83.2/13	87.5/15	89.3/16	88.4/16	84.2/9	89.5/13	89.5/17
		Fuzzy-Rough	**84.1/10**	84.1/13	84.1/12	**80.8/14**	**84.0/14**	**85.1/15**	**88.4/16**	**90.2/17**	**89.3/16**	**89.5/17**	**94.7/23**	**94.7/19**
Different Indices	Max Relevance	Class Sepr	60.3/7	57.2/6	59.2/7	55.5/6	46.1/5	58.2/7	75.0/11	76.8/12	77.7/13	68.4/12	73.7/13	84.2/15
		DB Index	77.2/10	75.7/11	77.1/10	72.0/9	76.0/11	77.8/10	77.7/11	79.5/13	79.5/14	63.2/10	57.9/9	47.4/8
		Dunn Index	56.0/7	49.2/6	50.6/7	55.5/7	47.7/7	57.8/8	80.4/13	81.3/14	79.5/14	63.2/10	56.4/8	45.6/8
		FFE Index	60.3/8	54.3/6	58.6/7	77.3/10	81.3/14	82.8/13	71.4/10	77.7/12	75.0/13	63.2/10	63.2/9	63.2/9
	Max Dependency	Class Sepr	60.3/7	57.2/7	58.6/6	52.6/5	46.1/5	57.9/6	75.0/9	75.0/12	76.8/11	67.5/11	73/12	82.8/13
		DB Index	79.5/10	75.3/9	78.9/11	68.4/8	73.7/10	78.9/11	78.6/10	75.0/10	79.5/11	62.9/9	57.6/9	47.1/7
		Dunn Index	56.0/7	51.8/8	50.6/6	55.5/7	47.4/6	57.9/7	71.4/9	77.7/12	78.6/10	63.1/11	55/8	44.6/7
		FFE Index	60.3/7	54.3/7	56.4/7	57.9/7	73.7/11	78.9/10	80.4/13	75.9/11	74.1/10	62.7/10	62.6/10	63/10
	MRMS	Class Sepr	60.3/7	57.2/7	59.2/7	55.5/7	46.1/5	58.2/7	75.0/11	76.8/11	77.7/12	68.4/11	73.7/10	84.2/11
		DB Index	83.3/10	81.2/9	83.4/11	72.0/9	76.0/11	77.8/11	80.4/12	81.3/13	79.5/12	63.2/10	57.9/9	47.4/8
		Dunn Index	56.0/8	49.2/6	50.6/7	55.5/9	47.7/6	57.8/9	80.4/12	81.3/13	79.5/13	63.2/10	56.4/9	45.6/8
		FFE Index	60.3/7	54.3/7	58.6/8	78.8/13	83.0/15	82.8/14	71.4/9	77.7/10	75.0/12	63.2/10	63.2/10	63.2/9
		Fuzzy-Rough	**84.1/10**	**84.1/13**	**84.1/12**	**80.8/14**	**84.0/14**	**85.1/15**	**88.4/16**	**90.2/17**	**89.3/16**	**89.5/17**	**94.7/23**	**94.7/19**
Different Methods	InfoGain		71.4/13	72.2/9	71.8/15	72.4/14	72.6/9	72.3/11	82.7/5	82.7/7	83.2/10	89.5/9	68.4/4	89.5/6
	Classical mRMR		75.5/9	75.4/10	75.4/10	72.8/10	73.8/10	74.3/10	84.8/11	84.8/10	84.8/11	84.2/12	84.2/13	89.5/13
	Fuzzy-Rough mRMR		84.0/11	**84.6/13**	83.7/12	80.3/11	84.1/12	84.7/12	87.5/13	89.3/12	**90.2/16**	89.5/13	89.5/14	94.7/15
	RELIEF		83.3/23	81.7/16	82.2/28	73.7/5	82.4/7	86.3/13	87.5/10	85.7/12	84.8/9	89.5/8	89.5/13	87.2/12
	SIMBA		83.2/22	81.5/21	82.7/20	74.8/3	84.3/9	84.3/11	87.5/13	89.3/15	85.7/12	**89.6/12**	84.2/13	85.3/11
	PCA		82.6/8	84.0/8	82.0/9	78.9/7	89.5/9	**94.7/14**	80.4/12	78.6/10	79.5/12	77.3/9	79.5/10	74.1/9
	ICA		83.3/10	83.5/12	82.2/11	75.1/9	90.0/13	90.1/11	85.2/14	85.7/11	84.8/15	89.5/14	89.5/13	84.2/10
	LDA		82.7/9	82.7/10	82.4/10	**82.0/12**	**90.3/14**	89.6/10	86.6/15	87.4/16	84.8/14	84.2/12	89.5/13	84.2/10
	Fuzzy-Rough MRMS		**84.1/10**	84.1/13	**84.1/12**	80.8/14	84.0/14	85.1/15	**88.4/16**	**90.2/17**	89.3/16	89.5/17	**94.7/23**	**94.7/19**

Table 3.2: Comparative Performance Analysis of Different Rough Sets on Colon, Breast I, and Lung

Different Classifiers	Different Criteria	Different Rough Sets	Colon				Breast I				Lung			
			Mean	StDv	Comp	d	Mean	StDv	Comp	d	Mean	StDv	Comp	d
K-NN	Max-Relevance	Classical	58.00	12.59	**3.45**	2	59.50	8.32	1.24	42	73.20	7.10	**2.91**	18
		Neighborhood	59.50	11.36	**3.36**	6	63.50	12.75	0.72	10	76.75	6.50	**1.91**	13
		Fuzzy	61.19	21.71	1.79	3	66.00	25.03	0.35	3	78.39	7.67	1.30	15
	Max-Dependency	Classical	59.53	24.54	1.81	4	59.50	8.32	1.24	3	77.31	8.45	1.53	8
		Neighborhood	69.52	14.75	0.90	2	61.50	10.34	0.98	11	79.47	8.94	0.90	4
		Fuzzy	68.10	11.18	1.42	2	63.00	10.34	0.81	11	79.47	8.45	0.93	6
	MRMS	Classical	59.50	8.32	**3.72**	38	59.50	8.32	1.24	3	80.61	7.10	0.67	13
		Neighborhood	63.09	12.59	**2.37**	7	62.00	8.32	0.95	13	80.61	8.42	0.62	11
		Fuzzy-Rough	74.29	8.00		3	70.00	25.39		44	82.84	7.67		43
SVM	Max-Relevance	Classical	64.52	2.62	**1.97**	10	57.00	10.34	1.30	8	74.50	7.10	**2.52**	12
		Neighborhood	65.00	9.37	1.59	11	57.50	12.75	1.08	10	76.17	9.25	1.76	13
		Fuzzy	67.61	24.93	0.66	42	59.00	18.86	0.64	4	77.84	7.67	1.46	18
	Max-Dependency	Classical	61.42	14.84	**1.87**	1	57.50	8.32	1.30	4	77.31	10.21	1.37	4
		Neighborhood	63.09	16.35	1.53	2	56.50	12.75	1.26	3	78.39	7.67	1.30	5
		Fuzzy	64.76	11.10	1.54	1	60.50	8.32	0.65	4	79.47	7.37	1.00	3
	MRMS	Classical	67.61	2.01	1.31	38	57.50	10.34	1.20	3	79.53	7.33	0.98	17
		Neighborhood	70.89	8.29	0.51	10	60.50	10.34	0.60	13	80.61	7.46	0.66	21
		Fuzzy-Rough	73.57	14.26		3	63.50	12.03		8	82.83	7.67		2
C4.5	Max-Relevance	Classical	63.09	21.27	1.35	11	61.50	17.51	0.91	14	75.06	9.63	**2.53**	11
		Neighborhood	64.52	15.89	1.42	8	60.00	15.81	1.14	13	76.75	7.98	**2.33**	18
		Fuzzy	66.19	11.65	1.38	15	62.50	17.51	0.79	15	78.39	10.21	1.51	21
	Max-Dependency	Classical	61.19	16.03	**1.94**	8	60.00	8.32	1.35	4	75.06	8.75	**2.71**	6
		Neighborhood	62.85	15.60	1.70	4	61.00	12.75	1.09	5	77.87	8.45	**1.90**	3
		Fuzzy	63.09	21.27	1.35	5	63.50	15.81	0.70	3	78.39	8.43	1.73	4
	MRMS	Classical	66.19	11.65	1.38	9	63.50	10.34	0.79	11	79.47	8.43	1.40	14
		Neighborhood	70.89	14.90	0.44	13	66.00	17.51	0.36	8	80.61	7.67	1.11	17
		Fuzzy-Rough	73.57	12.31		12	69.00	19.33		14	83.95	5.59		18

Table 3.3: Comparative Performance Analysis of Different Rough Sets on Leukemia I, Isolet, and Multiple Features

Different Classifiers	Different Criteria	Different Rough Sets	Leukemia I				Isolet				Multiple Features			
			Mean	StDv	Comp	d	Mean	StDv	Comp	d	Mean	StDv	Comp	d
K-NN	Max-Relevance	Classical	70.72	5.00	1.68	2	58.12	13.28	**5.23**	17	74.90	8.38	**5.24**	15
		Neighborhood	71.96	16.13	0.98	8	73.43	8.94	**2.58**	23	75.65	9.01	**4.67**	18
		Fuzzy	73.21	14.58	0.84	14	78.39	6.22	1.39	27	80.60	10.42	**2.66**	10
	Max-Dependency	Classical	69.29	17.45	1.32	5	58.12	14.92	**4.74**	9	76.10	8.03	**5.00**	7
		Neighborhood	70.36	16.13	1.22	7	78.94	11.94	0.76	6	74.60	12.73	**3.66**	4
		Fuzzy	71.96	16.92	0.95	6	79.71	6.38	0.88	4	80.40	9.22	**3.04**	6
	MRMS	Classical	71.96	19.85	0.86	7	67.09	13.22	**3.29**	17	86.10	8.31	1.31	17
		Neighborhood	74.64	16.13	0.58	12	79.17	11.32	0.73	13	87.40	9.03	0.79	12
		Fuzzy-Rough	78.57	13.88		7	82.12	5.83		44	89.80	3.26		31
SVM	Max-Relevance	Classical	65.00	23.63	0.55	8	58.07	14.21	**5.20**	21	81.10	8.93	**3.65**	12
		Neighborhood	66.43	18.38	0.46	10	74.17	11.92	**2.06**	18	80.20	10.25	**3.49**	9
		Fuzzy	66.61	16.71	0.46	9	78.34	7.31	1.53	12	82.80	9.21	**3.00**	17
	Max-Dependency	Classical	67.14	16.13	0.39	10	58.07	13.63	**5.41**	5	79.60	8.03	**4.55**	16
		Neighborhood	65.00	14.20	0.72	12	78.53	7.21	1.47	9	80.60	11.27	**3.09**	8
		Fuzzy	63.57	17.24	0.85	45	79.49	4.29	1.54	11	83.45	12.03	**2.19**	4
	MRMS	Classical	67.14	16.13	0.39	12	74.17	5.94	**3.62**	22	89.75	4.38	1.35	17
		Neighborhood	67.68	15.53	0.32	14	79.29	4.33	1.65	21	90.65	4.26	0.85	13
		Fuzzy-Rough	70.00	16.60		48	82.35	3.99		41	92.10	3.34		28
C4.5	Max-Relevance	Classical	69.29	17.45	0.89	9	57.98	16.25	**2.72**	8	76.10	6.93	**3.85**	15
		Neighborhood	70.36	17.17	0.75	14	64.65	12.66	**1.94**	8	78.20	10.73	**2.21**	19
		Fuzzy	71.96	16.13	0.56	11	65.06	11.04	**2.00**	7	79.55	9.22	**2.09**	9
	Max-Dependency	Classical	68.93	17.96	0.92	5	57.98	16.25	**2.72**	8	76.45	8.93	**3.10**	13
		Neighborhood	69.29	16.13	0.92	2	64.65	12.66	**1.94**	8	84.40	10.71	0.55	9
		Fuzzy	71.96	19.85	0.50	7	65.06	11.04	**2.00**	7	79.35	9.04	**2.18**	5
	MRMS	Classical	70.36	15.35	0.79	12	66.51	9.28	**1.84**	17	85.30	11.21	0.30	18
		Neighborhood	73.39	22.91	0.30	15	72.03	11.13	0.64	14	87.30	8.77	-0.27	8
		Fuzzy	76.07	16.82		16	75.32	11.94		38	86.45	4.93		19

3.4.2 Performance of Various Rough Set Based Models

In dimensionality reduction method, the reduced feature set is always relative to a certain feature evaluation index. In general, different evaluation indices may lead to different reduced feature subsets. To establish the effectiveness of fuzzy-rough sets over Pawlak's or classical and neighborhood rough sets, extensive experiments are done on various data sets. Different feature evaluation criteria such as Max-Dependency, Max-Relevance, and MRMS are considered for feature selection. In this regard, it should be mentioned that the classical or Pawlak's rough set based feature selection method reported in [327] uses the MRMS criterion, while the quick reduct [83], fuzzy-rough quick reduct [220], and neighborhood quick reduct [198] algorithms select features using Max-Dependency criterion.

Tables 3.1, 3.2, and 3.3 present the comparative performance of different rough set models for attribute selection task. The results and subsequent discussions are presented in these tables with respect to the classification accuracy of the K-NN, SVM, and C4.5. From the results reported in Table 3.1, it can be seen that the fuzzy-rough MRMS method attains maximum classification accuracy of the K-NN, SVM, and C4.5 in most of the cases. Out of 12 cases of training-testing, the fuzzy-rough MRMS method achieves highest classification accuracy in 10 cases, while neighborhood rough set based MRMS method attains it only in 2 cases.

On the other hand, among the 144 comparisons of 10-fold CV reported in Tables 3.2 and 3.3, the MRMS criterion with fuzzy-rough sets provides significantly better results in 47 cases and better but not significant in 96 cases, while better result is achieved only in 1 case using neighborhood rough sets based on the MRMS criterion. In brief, out of total 30 cases, the fuzzy-rough sets and neighborhood rough sets attain highest classification accuracy in 27 and 3 cases, respectively, based on the MRMS criterion.

Following conclusions can be drawn from the results reported in Tables 3.1, 3.2, and 3.3:

- the performance of the MRMS criterion is significantly better than that of other criteria, namely, Max-Dependency and Max-Relevance, irrespective of rough set models used;
- the performance of fuzzy-rough sets is significantly better than that of classical and neighborhood rough sets, irrespective of the feature evaluation criteria used; and
- the MRMS criterion based feature selection method using fuzzy-rough sets achieves higher classification accuracy in most of the cases, irrespective of the data sets, feature evaluation criteria, rough set models, experimental setup, and classifiers used.

The better performance of the FR-MRMS method is achieved due to the fact that the MRMS criterion can identify relevant and significant features from high dimensional real

life data sets more efficiently than Max-Dependency and Max-Relevance criteria, while the fuzzy-rough sets can capture uncertainties associated with the data more accurately.

3.4.3 Performance of Various Feature Evaluation Indices

In order to establish the effectiveness of fuzzy-rough sets over other feature evaluation indices, such as class separability index, DB index, Dunn index, and FFEI, extensive experimentation is done on different real life data sets. Tables 3.1, 3.4, and 3.5 present the comparative performance of FR-MRMS method and various feature evaluation indices considering different feature evaluation criteria such as Max-Relevance, Max-Dependency, and MRMS.

From the results reported in Table 3.1, it can be seen that the FR-MRMS method attains highest classification accuracy on Satimage, Segmentation, Leukemia II, and Breast II data sets, irrespective of the classifiers used. Tables 3.4 and 3.5 report the comparative performance in case of 10-fold CV on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features data sets. The results and subsequent discussions are analyzed in these tables with respect to the classification accuracy of the K-NN, SVM, and C4.5. All the results reported in Tables 3.4 and 3.5 confirm that the FR-MRMS method provides significantly better results in 94 cases and better but not significantly in 120 cases out of total 216 cases. On the other hand, both DB index and FFEI based on the MRMS criterion achieve better results, but not significantly, than that of the proposed FR-MRMS method in only 1 case each. The results reported in Tables 3.1, 3.4, and 3.5 also establish the fact that the performance of the MRMS criterion is better than that of other two criteria, namely, Max-Dependency and Max-Relevance, irrespective of feature evaluation indices, classifiers, and data sets used.

3.4.4 Performance of Different Algorithms

Finally, Tables 3.1, 3.6, and 3.7 compare the performance of the proposed FR-MRMS algorithm with that of different existing feature selection and extraction algorithms on various data sets. From the results reported in Table 3.1, it is seen that the FR-MRMS algorithm achieves highest classification accuracy of the K-NN, SVM, and C4.5 in 6 cases out of total 12 cases, while the fuzzy-rough mRMR, SIMBA, PCA, and LDA attain highest classification accuracy in only 2, 1, 1, and 2 cases, respectively. Tables 3.6 and 3.7 report the performance of different methods in case of 10-fold CV, along with the results of test of significance, for the K-NN, SVM, and C4.5. From the results reported in these tables, it can be seen that the proposed method attains significantly better results than other algorithms in 28 cases out of total 144 cases and better but not significant in 109 cases, while better but not significant results are achieved by the ICA and LDA in 4 and 3

Table 3.4: Comparative Performance Analysis of Different Indices on Colon, Breast I, and Lung

Different Classifiers	Different Criteria	Different Indices	Colon				Breast I				Lung			
			Mean	StDv	Comp	d	Mean	StDv	Comp	d	Mean	StDv	Comp	d
K-NN	Max Relevance	Class Separability	68.24	13.29	1.23	12	58.50	8.32	1.36	11	75.06	8.85	**2.10**	10
		DB Index	70.95	14.67	0.63	2	59.00	11.65	1.25	12	76.75	7.10	**1.84**	13
		Dunn Index	71.18	4.53	1.07	4	59.50	13.01	1.16	7	76.75	8.01	1.73	16
		FFEI	69.62	11.24	1.07	13	59.00	11.65	1.48	10	75.06	7.56	**2.28**	14
	Max Dependency	Class Separability	65.21	14.22	1.76	3	57.50	8.32	1.33	4	75.06	7.55	**2.29**	5
		DB Index	66.83	11.25	1.71	2	59.00	12.48	1.23	2	75.06	7.55	**2.29**	4
		Dunn Index	66.83	11.25	1.71	4	58.50	10.34	1.33	3	77.87	6.50	1.56	6
		FFEI	65.38	10.66	**2.11**	3	58.00	12.48	1.34	3	75.06	7.13	**2.35**	10
	MRMS	Class Separability	72.62	10.95	0.39	5	57.50	10.34	1.44	10	76.75	7.83	1.75	11
		DB Index	72.38	13.84	0.38	3	61.50	10.34	0.98	16	76.75	5.59	**2.03**	10
		Dunn Index	72.38	13.84	0.38	4	61.50	12.48	0.95	13	77.87	10.21	1.23	14
		FFEI	70.95	11.46	0.75	7	59.00	13.01	1.22	18	76.17	7.67	**1.95**	18
		Fuzzy-Rough	74.29	8.00		3	70.00	25.39		44	82.84	7.67		43
SVM	Max Relevance	Class Separability	68.81	14.61	0.74	8	57.00	12.75	1.17	13	74.50	10.55	**2.02**	15
		DB Index	70.71	15.51	0.43	13	57.50	12.75	1.08	17	76.17	9.80	1.70	17
		Dunn Index	70.71	15.51	0.43	14	57.50	18.86	0.85	12	76.72	8.85	1.65	19
		FFEI	69.05	14.89	0.69	17	57.00	12.75	1.17	16	75.06	7.67	**2.27**	10
	Max Dependency	Class Separability	67.38	16.07	0.91	18	56.50	12.48	1.28	3	75.06	8.43	**7.00**	5
		DB Index	68.81	14.61	0.74	12	56.00	13.01	1.34	4	76.17	10.55	1.62	6
		Dunn Index	69.05	14.89	0.69	9	57.00	12.75	1.17	2	77.31	9.80	1.40	3
		FFEI	67.38	13.84	0.98	10	56.00	8.32	1.62	5	75.06	8.43	**2.16**	6
	MRMS	Class Separability	72.14	11.12	0.25	10	56.50	8.32	1.51	10	75.61	8.85	**1.95**	14
		DB Index	72.38	13.84	0.19	4	58.00	12.48	1.00	17	76.75	7.67	1.77	15
		Dunn Index	72.38	13.84	0.19	4	57.50	8.32	1.30	5	77.87	7.60	1.46	21
		FFEI	71.90	13.96	0.26	6	57.00	13.01	1.16	6	76.17	8.43	**1.85**	13
		Fuzzy-Rough	73.57	14.26		3	63.50	12.03		8	82.83	7.67		2
C4.5	Max Relevance	Class Separability	69.76	12.14	0.70	14	60.50	12.75	1.16	13	75.06	8.01	**2.88**	13
		DB Index	70.95	12.89	0.46	13	63.50	8.32	0.83	11	76.17	7.56	**2.62**	11
		Dunn Index	70.95	12.14	0.48	9	63.50	11.65	0.77	12	77.87	6.50	**2.24**	20
		FFEI	69.05	11.40	0.85	3	61.00	8.32	1.20	7	75.06	7.13	**3.10**	17
	Max Dependency	Class Separability	66.19	14.05	1.25	4	59.50	12.48	1.31	9	75.61	8.01	**2.70**	6
		DB Index	67.38	10.83	1.19	3	58.00	13.01	1.49	4	77.87	7.56	**2.05**	8
		Dunn Index	68.09	11.18	1.04	5	58.50	12.75	1.43	5	77.87	8.42	**1.90**	4
		FFEI	66.19	12.59	1.33	3	59.50	8.32	1.43	3	75.06	7.97	**2.89**	6
	MRMS	Class Separability	71.90	14.05	0.28	10	63.00	8.32	0.90	10	77.87	7.83	**2.00**	14
		DB Index	72.38	15.59	0.19	13	65.00	12.48	0.55	13	78.42	5.59	**2.21**	18
		Dunn Index	72.38	13.22	0.21	5	64.50	12.48	0.62	14	78.42	8.36	1.74	24
		FFEI	71.90	14.05	0.28	13	63.50	10.34	0.79	11	76.17	8.85	**2.35**	18
		Fuzzy-Rough	73.57	12.31		12	69.00	19.33		14	83.95	5.59		18

Table 3.5: Comparative Performance Analysis of Different Indices on Leukemia I, Isolet, and Multiple Features

Different Classifiers	Different Criteria	Different Indices	Leukemia I				Isolet				Multiple Features			
			Mean	StDv	Comp	d	Mean	StDv	Comp	d	Mean	StDv	Comp	d
K-NN	Max Relevance	Class Separability	67.50	18.30	1.52	12	45.33	16.93	**6.50**	11	72.60	10.71	**4.86**	11
		DB Index	68.93	20.33	1.24	10	73.26	12.33	**2.06**	14	73.90	8.31	**5.63**	18
		Dunn Index	68.93	14.20	1.53	15	73.11	11.57	**2.20**	12	73.90	9.03	**5.24**	9
		FFEI	67.68	15.53	1.65	13	63.88	10.34	**4.86**	16	73.20	9.26	**5.35**	14
	Max Dependency	Class Separability	66.61	16.71	1.74	9	45.09	16.22	**6.79**	7	71.90	9.26	**5.77**	5
		DB Index	67.86	19.07	1.44	7	68.17	12.64	**3.17**	6	73.90	9.03	**5.24**	8
		Dunn Index	69.29	17.45	1.32	8	69.42	12.43	**2.93**	6	72.70	8.38	**6.01**	4
		FFEI	67.86	17.26	1.53	6	63.88	10.93	**4.66**	8	72.40	9.22	**5.63**	6
	MRMS	Class Separability	71.96	19.85	0.86	13	45.20	17.53	**6.32**	15	80.40	9.03	**3.10**	17
		DB Index	73.21	20.22	0.69	10	73.11	10.38	**2.40**	21	82.05	10.71	**2.19**	13
		Dunn Index	74.64	20.22	0.51	12	72.89	9.37	**2.65**	16	82.10	8.38	**2.71**	23
		FFEI	70.36	16.13	1.22	14	63.88	11.26	**4.55**	19	80.60	8.31	**3.26**	18
		Fuzzy-Rough	78.57	13.88		7	82.12	5.83		44	89.80	3.26		31
SVM	Max Relevance	Class Separability	60.71	18.13	1.19	13	45.25	15.65	**7.27**	13	76.90	12.27	**3.78**	12
		DB Index	63.75	21.59	0.72	16	73.23	11.52	**2.37**	16	80.10	9.22	**3.87**	14
		Dunn Index	63.75	16.13	0.85	10	72.62	10.25	**2.80**	19	79.95	10.71	**3.42**	21
		FFEI	61.96	16.95	1.07	14	71.01	11.77	**2.89**	10	78.95	8.31	**4.64**	19
	Max Dependency	Class Separability	62.68	16.95	0.98	6	45.25	16.21	**7.03**	6	78.40	8.38	**4.80**	6
		DB Index	64.82	14.36	0.74	4	69.74	9.32	**3.93**	8	80.30	7.22	**4.69**	3
		Dunn Index	64.82	17.29	0.68	3	69.80	8.31	**4.31**	4	80.20	9.03	**3.91**	8
		FFEI	62.68	14.36	0.00	4	71.01	8.33	**3.88**	6	79.15	9.01	**4.26**	4
	MRMS	Class Separability	64.82	17.29	0.68	13	45.30	18.21	**6.29**	11	80.70	8.31	**4.03**	18
		DB Index	65.18	15.35	0.67	18	73.37	8.21	**3.11**	17	85.30	10.71	**1.92**	9
		Dunn Index	65.18	19.98	0.59	11	72.72	7.66	**3.53**	21	85.35	9.26	**2.17**	32
		FFEI	63.93	17.53	0.80	14	69.87	8.52	**4.19**	22	82.50	13.27	**2.22**	17
		Fuzzy-Rough	70.00	16.60		48	82.35	3.99		41	92.10	3.34		28
C4.5	Max Relevance	Class Separability	67.50	16.92	1.14	14	45.43	12.53	**5.46**	21	74.30	9.01	**3.74**	16
		DB Index	68.93	17.45	0.93	11	62.60	10.33	**2.55**	13	77.35	7.22	**3.29**	19
		Dunn Index	68.93	19.85	0.87	8	54.93	11.43	**3.90**	18	77.40	9.03	**2.78**	11
		FFEI	67.68	17.17	1.11	15	52.01	9.42	**4.85**	16	76.70	8.38	**3.17**	10
	Max Dependency	Class Separability	65.00	20.55	1.32	8	45.43	13.28	**5.29**	7	73.75	11.27	**3.26**	5
		DB Index	67.86	16.13	1.12	6	60.84	14.27	**2.46**	6	75.80	10.71	**2.86**	8
		Dunn Index	68.93	16.13	0.97	5	54.93	11.84	**3.83**	18	76.25	9.26	**3.07**	4
		FFEI	66.61	16.71	1.26	7	52.40	12.53	**4.19**	6	75.20	8.31	**3.68**	10
	MRMS	Class Separability	70.36	16.13	0.78	9	45.52	16.27	**4.67**	12	83.30	9.26	0.95	19
		DB Index	71.96	19.85	0.50	10	63.92	9.27	**2.38**	31	87.15	7.22	-0.25	8
		Dunn Index	71.96	17.83	0.53	11	54.93	10.22	**4.10**	18	86.20	8.38	0.08	17
		FFEI	70.36	16.71	0.76	12	52.40	11.23	**4.42**	6	87.30	9.01	-0.26	21
		Fuzzy-Rough	76.07	16.82		16	75.32	11.94		38	86.45	4.93		19

Table 3.6: Comparative Performance Analysis of Different Methods on Colon, Breast I, and Lung

Different Classifiers	Different Methods / Algorithms	Colon				Breast I				Lung			
		Mean	StDv	Comp	d	Mean	StDv	Comp	d	Mean	StDv	Comp	d
K-NN	InfoGain	67.86	17.82	1.04	2	56.00	27.97	1.17	7	68.10	11.18	**3.44**	37
	Classical mRMR	67.86	16.12	1.13	46	59.50	18.33	1.06	6	68.10	11.18	**3.44**	37
	Fuzzy-Rough mRMR	71.43	20.26	0.42	5	56.00	20.66	1.35	3	82.84	7.67	0.00	37
	RELIEF	65.71	15.07	1.59	6	59.50	8.32	1.24	13	74.53	7.62	**2.43**	17
	SIMBA	66.19	14.89	1.51	14	61.00	9.83	1.05	16	76.75	9.07	1.62	19
	PCA	68.88	15.24	0.99	4	63.50	14.38	0.70	9	79.47	10.21	0.83	11
	ICA	67.68	16.82	1.12	7	63.50	15.93	0.69	12	80.61	8.45	0.62	10
	LDA	67.86	11.35	1.46	10	63.00	13.37	0.77	9	80.06	8.43	0.77	13
	Fuzzy-Rough MRMS	74.29	8.00		3	70.00	25.39		44	82.84	7.67		43
SVM	InfoGain	68.33	16.91	0.75	14	60.00	28.28	0.36	12	67.25	34.12	1.41	49
	Classical mRMR	72.38	11.40	0.21	11	58.00	33.27	**3.22**	9	76.20	8.85	1.79	17
	Fuzzy-Rough mRMR	73.09	19.38	0.06	10	62.00	23.94	**9.46**	26	77.41	8.45	1.50	19
	RELIEF	65.28	12.31	1.39	10	59.50	7.55	0.89	10	75.09	8.07	**2.20**	13
	SIMBA	67.01	11.23	1.14	13	59.50	8.26	0.87	9	75.64	8.04	**2.05**	10
	PCA	68.24	10.63	0.95	11	62.00	13.37	0.26	9	77.31	7.70	1.61	10
	ICA	66.97	11.44	1.14	6	61.00	13.70	0.43	7	76.20	7.53	**1.95**	9
	LDA	67.56	9.89	1.09	9	62.00	9.43	0.31	8	77.41	9.03	1.45	12
	Fuzzy-Rough MRMS	73.57	14.26		3	63.50	12.03		8	82.83	7.67		2
C4.5	InfoGain	66.19	11.65	1.38	12	57.50	14.29	1.51	13	68.01	11.70	**3.89**	9
	Classical mRMR	72.38	11.40	0.22	11	59.00	11.32	1.41	12	77.86	8.35	**1.91**	11
	Fuzzy-Rough mRMR	73.09	16.30	0.07	9	57.50	15.94	1.45	14	78.39	8.94	1.66	15
	RELIEF	64.76	18.02	1.28	13	59.00	13.99	1.33	11	75.64	7.37	**2.84**	12
	SIMBA	66.19	19.56	1.01	14	60.50	12.03	1.18	12	76.20	8.45	**2.42**	15
	PCA	69.76	12.14	0.70	11	63.50	22.74	0.58	6	78.98	7.83	1.63	6
	ICA	68.24	13.57	0.92	9	63.00	23.52	0.62	7	77.86	7.46	**2.06**	7
	LDA	69.76	17.04	0.57	9	61.50	16.47	0.93	12	78.98	7.56	1.67	9
	Fuzzy-Rough MRMS	73.57	12.31		12	69.00	19.33		14	83.95	5.59		18

Table 3.7: Comparative Performance Analysis of Different Methods on Leukemia I, Isolet, and Multiple Features

Different Classifiers	Different Methods / Algorithms	Leukemia I				Isolet				Multiple Features			
		Mean	StDv	Comp	d	Mean	StDv	Comp	d	Mean	StDv	Comp	d
K-NN	InfoGain	72.14	18.83	0.87	20	76.09	11.38	1.49	26	76.70	5.03	**6.91**	21
	Classical mRMR	70.89	16.79	1.11	18	77.04	9.11	1.49	15	79.75	9.37	**3.20**	14
	Fuzzy-Rough mRMR	73.22	4.53	1.16	10	82.06	6.64	0.03	23	83.75	10.21	1.79	13
	RELIEF	71.96	16.71	0.96	10	78.32	7.10	1.31	16	81.80	8.38	**2.81**	16
	SIMBA	73.39	17.26	0.74	11	78.38	4.87	1.56	11	82.90	8.27	**2.45**	13
	PCA	74.64	17.83	0.55	8	76.21	5.38	**2.36**	10	84.15	7.22	**2.26**	8
	ICA	74.64	14.00	0.63	12	78.27	6.44	1.40	13	85.10	8.05	1.71	11
	LDA	76.07	16.82	0.36	10	78.43	5.18	1.50	14	84.65	9.21	1.67	9
	Fuzzy-Rough MRMS	78.57	13.88		7	82.12	5.83		44	89.80	3.26		31
SVM	InfoGain	60.00	28.28	0.96	12	76.47	13.83	1.29	16	79.50	10.32	**3.67**	16
	Classical mRMR	65.00	18.58	0.63	16	75.63	12.48	1.62	12	83.10	9.26	**2.89**	15
	Fuzzy-Rough mRMR	68.57	19.98	0.17	49	81.92	8.33	0.15	21	88.60	8.31	1.24	21
	RELIEF	66.43	19.57	0.44	10	79.18	8.19	1.10	14	83.90	10.33	**2.39**	15
	SIMBA	67.68	14.00	0.34	21	79.43	7.33	1.11	13	85.20	9.51	**2.16**	13
	PCA	69.29	17.45	0.09	11	80.76	8.82	0.52	9	85.70	8.05	**2.32**	7
	ICA	70.36	16.13	-0.05	9	83.25	5.03	-0.44	7	84.95	10.27	**2.09**	11
	LDA	70.71	16.92	-0.10	14	82.07	6.18	0.12	12	85.75	8.21	**2.27**	9
	Fuzzy-Rough MRMS	70.00	16.60		48	82.35	3.99		41	92.10	3.34		28
C4.5	InfoGain	70.36	16.13	0.78	7	71.48	9.38	0.80	14	78.50	9.26	**2.40**	19
	Classical mRMR	70.54	20.91	0.65	8	71.54	10.42	0.75	19	80.35	10.44	1.67	21
	Fuzzy-Rough mRMR	73.21	19.85	0.35	11	73.48	7.44	0.42	22	81.05	9.04	1.66	9
	RELIEF	71.79	20.49	0.51	9	71.75	6.06	0.85	18	80.80	9.26	1.70	14
	SIMBA	71.96	19.85	0.50	10	72.12	7.27	0.73	11	81.30	8.38	1.68	16
	PCA	73.39	22.91	0.30	8	66.88	8.22	**1.84**	12	85.20	11.28	0.32	11
	ICA	74.64	17.83	0.18	7	80.76	6.74	-1.25	9	87.80	9.22	-0.41	9
	LDA	74.64	14.00	0.21	7	82.25	6.11	-1.63	14	88.90	5.27	-1.07	10
	Fuzzy-Rough MRMS	76.07	16.82		16	75.32	11.94		38	86.45	4.93		19

cases, respectively. In brief, out of total 30 cases, the FR-MRMS method attains highest classification accuracy in 20 cases, while the fuzzy-rough mRMR, SIMBA, PCA, ICA, and LDA achieve it only 2, 1, 1, 1, and 5 cases, respectively.

All the results reported in Tables 3.1, 3.6, and 3.7 also establish the fact that the mRMR criterion based feature selection method in fuzzy approximation spaces (fuzzy-rough mRMR) [323] improves the classification accuracy significantly over its crisp counterpart (classical mRMR) [416], irrespective of the classifiers and data sets used. Out of total 30 cases, the fuzzy-rough mRMR method provides better accuracy than that of classical mRMR in 28 cases. Only for Breast Cancer I data set, the classical mRMR performs better with respect to both K-NN and C4.5. Moreover, Table 3.8 reports the execution time of different algorithms. The significantly lesser time of the proposed algorithm is achieved due to its low computational complexity.

Hence, all the results reported in Tables 3.1, 3.6, and 3.7 confirm that the proposed FR-MRMS method selects a set of features having highest classification accuracy of the K-NN, SVM, and C4.5 in most of the cases, irrespective of the data sets. Also, the proposed method can potentially yield significantly better results than the existing algorithms. The better performance of the proposed method is achieved due to the fact that it provides an efficient way to select a reduced set of features having maximum relevance and significance.

3.5 Conclusion

The dimensionality reduction by attribute selection is one of the important problems in pattern recognition, machine learning, and data mining, particularly given the explosive growth of available information. In this regard, the contribution of this chapter is three fold, namely,

1. development of a new feature selection method, termed as FR-MRMS, integrating judiciously the theory of fuzzy-rough sets and merits of the MRMS criterion;

2. application of the proposed method in selecting discriminative and significant features from high dimensional benchmark and microarray gene expression data sets; and

3. compare the performance of the proposed method with that of some existing methods using the predictive accuracy of three classifiers, namely, nearest neighbor rule, decision tree, and support vector machine.

The proposed FR-MRMS method uses the concept of fuzzy-rough feature relevance and significance for finding relevant and significant features of real valued data sets. This

Table 3.8: Execution Time (in second) of Different Methods for Various Data Sets

Different Criteria	Different Indices	Different Benchmark Data Sets									
		Satimage	Segmentation	Leukemia II	Breast II	Colon	Breast I	Lung	Leukemia I	Isolet	Mult.Feat.
Max Relevance	Classical Rough	0.10	0.10	5.21	3.40E+01	0.20	0.31	2.00E+01	0.80	8.30	3.30
	Neighborhood	8.40E+02	6.30E+02	5.70E+04	1.50E+05	5.40E+02	6.30E+02	8.10E+04	7.30E+02	2.10E+04	1.00E+04
	Fuzzy-Rough	1.70E+01	1.60E+01	2.20E+01	4.50E+01	0.20	2.21	8.41	2.10	2.20E+01	7.41
	Class Separability	0.08	0.04	6.28	1.41E+01	0.26	0.38	2.77	0.21	1.26	0.28
	DB Index	0.12	0.03	2.10	4.70	0.28	0.39	1.28	0.32	1.66	0.48
	Dunn Index	0.11	0.03	2.08	4.77	0.19	0.22	0.96	0.28	1.72	0.37
	FFE Index	2.52E+03	0.04	7.25E+03	1.64E+04	2.98E+03	3.43E+03	9.83E+03	3.83E+03	1.83E+04	4.78E+03
Max Dependency	Classical Rough	4.91	3.30	9.40E+02	3.30E+02	2.60E+01	1.40E+01	8.30E+01	1.80E+01	1.30E+02	4.60E+01
	Neighborhood	2.40E+04	2.80E+04	5.30E+05	1.80E+06	1.20E+04	9.30E+03	6.20E+05	1.10E+04	2.20E+05	6.70E+04
	Fuzzy-Rough	3.30E+04	2.70E+04	5.50E+05	1.30E+06	9.80E+03	9.80E+03	5.90E+05	9.40E+03	2.60E+05	5.20E+04
	Class Separability	0.45	0.05	7.68	1.71E+01	1.77	1.98	3.22	2.04	4.28	2.83
	DB Index	0.84	0.04	2.98	6.59	2.34	1.82	3.82	2.16	5.02	3.92
	Dunn Index	0.78	0.04	3.12	6.96	1.92	1.55	3.21	1.93	4.88	4.27
	FFE Index	8.34E+03	0.06	8.25E+03	1.85E+04	1.19E+04	1.43E+04	1.38E+04	1.27E+04	1.87E+04	1.53E+04
MRMS	Classical Rough	0.31	0.24	7.90	3.80E+01	0.20	0.40	4.30E+01	1.52	9.40	5.92
	Neighborhood	1.00E+03	6.50E+02	8.70E+04	2.80E+05	7.30E+02	8.20E+02	9.20E+04	9.20E+02	3.40E+04	2.10E+04
	Class Separability	0.46	0.25	6.12E+02	4.80E+01	0.52	0.62	3.82	0.42	1.52	0.42
	DB Index	2.09	0.22	1.12E+03	2.50E+03	0.37	0.62	2.55	0.41	2.81	0.61
	Dunn Index	2.62	0.20	6.01E+02	1.37E+03	0.29	0.44	1.28	0.45	3.12	0.52
	FFE Index	5.33E+04	4.25E+01	7.53E+04	1.73E+05	4.32E+03	4.84E+03	1.23E+04	5.27E+03	2.13E+04	6.72E+03
InfoGain		0.09	0.05	4.72	1.05E+01	0.51	5.39	1.71E+01	5.40	7.02	0.36
Classical mRMR		0.10	0.10	5.00	5.40	0.52	5.51	1.80E+01	5.40	7.40	0.40
Fuzzy-Rough mRMR		1.20	1.80	6.50	5.60	1.71	6.10E+01	3.80E+01	3.10E+01	5.60E+01	1.40E+01
RELIEF		2.77	1.29	4.22	9.70	2.06	5.28E+01	2.17E+01	1.83E+01	3.26E+01	1.24E+01
SIMBA		1.02E+03	3.82E+02	4.11E+03	9.15E+03	3.39E+02	5.92E+02	6.83E+04	4.59E+02	3.02E+03	2.01E+03
PCA		7.80E+03	8.00E+02	4.00E+03	4.00E+04	2.20E+04	3.00E+04	3.20E+04	3.80E+04	1.10E+04	1.80E+04
ICA		1.23E+03	4.51E+02	5.63E+03	1.28E+04	6.29E+02	7.22E+02	8.47E+02	5.29E+02	4.11E+03	3.11E+03
LDA		9.81E+02	4.92E+02	8.92E+02	1.97E+03	3.18E+02	5.11E+02	1.13E+03	4.18E+02	8.31E+02	2.18E+03
Fuzzy-Rough MRMS		4.30E+01	2.30E+01	2.70E+01	6.08E+01	0.40	2.81	9.52	2.71	3.20E+01	1.00E+01

formulation is geared towards maximizing the utility of fuzzy-rough sets, feature selection, and the MRMS criterion with respect to knowledge discovery tasks. Through these investigations and experiments, the potential utility of fuzzy-rough sets and the MRMS criterion for attribute selection is demonstrated.

Traditional fuzzy-rough sets, also known as type-1 fuzzy-rough sets, is useful for feature selection in uncertain environment. But, it fails to provide good performance when determination of exact membership function for any fuzzy set is difficult. The type-1 fuzzy set was generalized to type-2 fuzzy set by Zadeh [555]. In this regard, next chapter presents a feature selection method, based on interval type-2 fuzzy-rough sets.

Chapter 4

Interval Type-2 Fuzzy-Rough Sets for Attribute Selection

4.1 Introduction

Dimensionality reduction, which has already been introduced in Chapter 3, is one of the important issues that has retained high interest in pattern recognition, machine learning, and data mining. In general, the high dimensional real life data sets contain a large number of irrelevant and redundant or insignificant attributes, which must be removed from the data set by retaining the optimum salient characteristics necessary for the pattern recognition process. Hence, the main objective of dimensionality reduction task is to select relevant and nonredundant features.

One of the main problems in real life high dimensional data analysis is uncertainty. In this regard, the possibility concept introduced by rough set theory becomes popular in modeling and propagating uncertainty. It is a very useful technique to find the data dependencies and hence can be used for selecting most informative feature subset of original features from a data set having discretized attributes [83, 327]. On the other hand, fuzzy set [150, 436, 438] theory permits the gradual assessment of the membership of elements in a set, which is an extension of the classical set. The theory of fuzzy-rough sets is developed to encapsulate two related and complementary, but distinct concepts generated from uncertainty in knowledge, vagueness (for fuzzy set) and indiscernibility concept (for rough sets), which can be used for feature selection of a data set having both discrete and continuous attributes [220, 311, 315, 323, 324].

Although the researchers found some limitations of conventional type-1 (T1) fuzzy sets, most of the research works in this domain are still restricted to this theory. The T1 fuzzy set was generalized to type-2 (T2) [171, 206, 232, 240, 343–346] fuzzy set by Zadeh [555]. In T2 fuzzy set, there are grades of membership which are fuzzy. This set can be useful

in such a situation where determination of exact membership function for any fuzzy set is difficult. According to Mendel [342], there are at least four sources of uncertainties in T1 fuzzy logic systems: (i) the meanings of the words that are used in the antecedents and consequents of rules can be uncertain; (ii) consequents may have a histogram of values associated with them, especially when knowledge is extracted from a group of experts who do not all agree; (iii) measurements that activate a T1 fuzzy logic system may be noisy and therefore uncertain; and (iv) the data that are used to tune the parameters of a T1 fuzzy logic system may also be noisy. All of these uncertainties translate into uncertainties about fuzzy set membership functions.

The T2 fuzzy set theory can be combined with rough set theory to construct the T2 fuzzy-rough sets, which is more useful where the exact membership function is not known. Three-dimensional membership function of the T2 fuzzy sets has more degrees of freedom than that of the T1 fuzzy system with two-dimensional membership function. Hence, the T2 fuzzy-rough sets can be a useful technique for attribute reduction. A T2 fuzzy-rough quick reduct algorithm based on Max-Dependency criterion has been proposed by Wu et al. [531]. According to Liang and Mendel [283] and Mendel et al. [345], the interval type-2 (IT2) fuzzy set is useful in practical applications as the computational complexity of the IT2 fuzzy set is lower than that of the T2 fuzzy set.

In this regard, an IT2 fuzzy-rough feature selection method (IT2FR-MRMS) is presented in this chapter, integrating judiciously the merits of IT2 fuzzy set, rough sets, and maximal relevance-maximal significance (MRMS) criterion, which can be used to reduce the real valued noisy features effectively. A subset of condition attributes is selected from the original feature set by maximizing the relevance and significance of the selected features. Both relevance and significance of the features are computed using the concept of IT2 fuzzy positive regions of IT2 fuzzy-rough sets. The concept of lower and upper fuzzy equivalence partition matrices is introduced to compute both relevance and significance in IT2 fuzzy approximation spaces. The generalized π function in the one dimensional form is used to generate fuzzy information granules corresponding to each condition attribute, where the centers and radii of the π functions can be determined automatically from the distribution of training patterns. The IT2 fuzzy positive regions of decision attributes or class labels are computed based on the concept of the IT2 fuzzy equivalence partition matrix. The effectiveness of the proposed IT2 fuzzy-rough attribute selection method, along with a comparison with other methods, is demonstrated on several benchmark and microarray gene expression data sets. Some of the results, presented in this chapter, are also reported in [316].

The structure of rest of this chapter is as follows: Section 4.2 briefly describes the basic notions of type-2 and interval type-2 fuzzy-rough sets. The proposed IT2 based fuzzy-rough attribute selection method is presented in Section 4.3. A few case studies and

a comparison with other methods are reported in Section 4.4. Concluding remarks are given in Section 4.5.

4.2 Basics of Type-2 Fuzzy-Rough Sets

In this section, the basic notions in the theories of T2 fuzzy sets are reported, along with IT2 fuzzy sets and IT2 fuzzy-rough sets.

4.2.1 T2 Fuzzy Sets

The memberships of the objects in a class along any feature axis can be represented using the normalized histogram plot (Fig. 4.1(a), 4.1(b)). The T2 fuzzy membership function is useful in the situation where the system cannot be easily modeled using any T1 membership function (Fig. 4.2(a)). The membership function of the T2 fuzzy set can be considered as the blurring of T1 membership function (Fig. 4.2(b)) by shifting the points either to the left or to the right. The shifting may be of different amount. In that case, for a particular value of x, the membership function does not have a single value, instead, it takes values from a band of distribution of points. This distribution provides a T2 three dimensional membership function which is the basis of T2 fuzzy set.

Definition 4.1 *Let $\mathbb{X}$ be the universe of discourse for the primary variable x, $\mathbb{U}$ be the universe of discourse for the secondary variable u, both of them are nonempty finite sets and assumed to be $[0, 1]$. A T2 fuzzy set, denoted as $\widetilde{A}$ (Fig. 4.3(a)), is represented by a T2 membership function $\mu_{\widetilde{A}}(x, u)$ [347], where $x \in \mathbb{X}$ and $u \in [0, 1]$, that is,*

$$\widetilde{A} = \{((x, u), \mu_{\widetilde{A}}(x, u))|x \in \mathbb{X}, u \in [0, 1]\}. \tag{4.1}$$

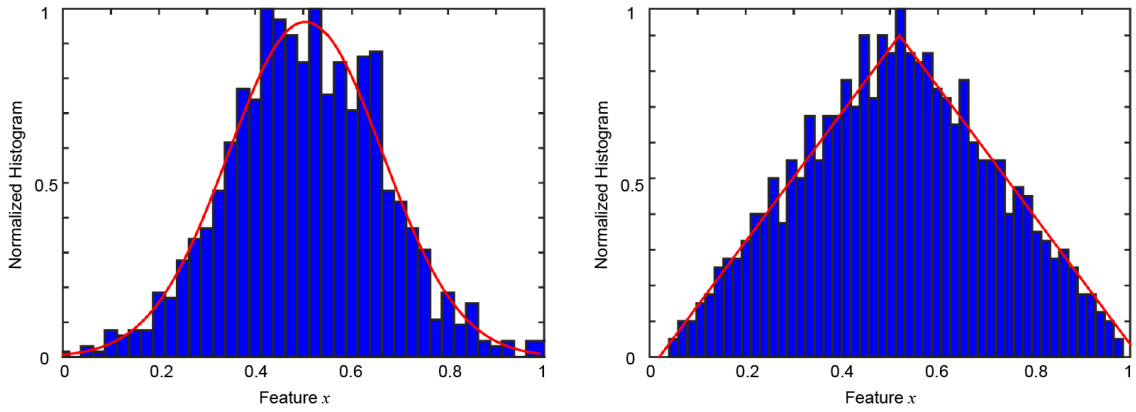
$\widetilde{A}$ may also be represented in integral form as

$$\begin{aligned} \widetilde{A} &= \int_{x\in\mathbb{X}} \int_{u\in J_x} \mu_{\widetilde{A}}(x, u)/(x, u) \\ &\equiv \int_{x\in\mathbb{X}} \Big[\int_{u\in J_x} f_x(u)/u \Big] \Big/ x, J_x \subseteq [0, 1]; \end{aligned} \tag{4.2}$$

where $f_x(u) = \mu_{\mathbb{X}}(x, u)$. The double integral denotes the union over all admissible x and u. Here, J_x can be represented as

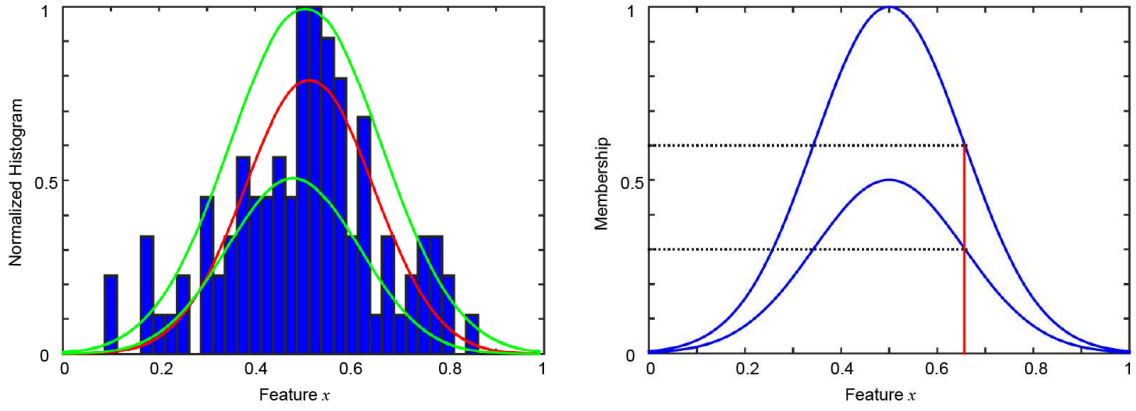
$$J_x = \{((x, u)|u \in [0, 1], \mu_{\widetilde{A}}(x, u)) > 0\}. \tag{4.3}$$

The constraint $u \in [0, 1]$ comes from the T1 fuzzy set which is same as $0 \leq \mu_{\widetilde{A}}(x) \leq 1$, that means, when uncertainty disappears in the T2 fuzzy membership function, the set



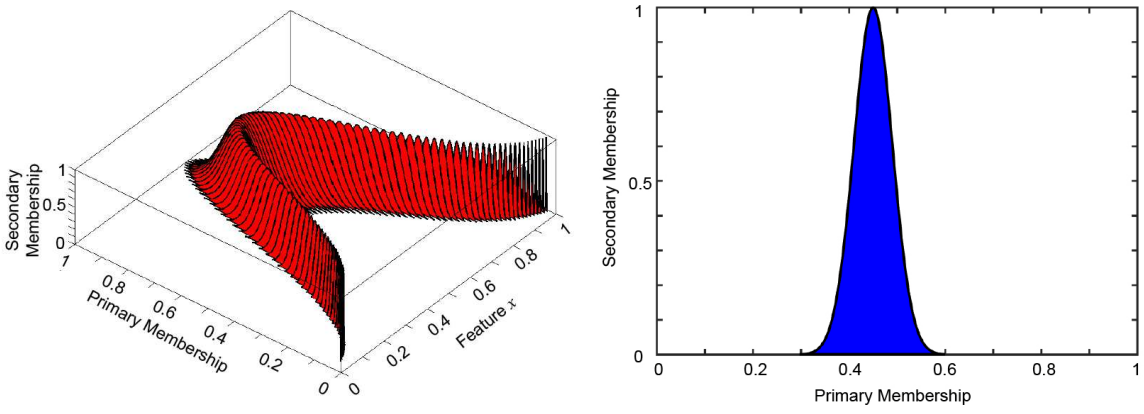
(a) (b)

Figure 4.1: Examples of membership function (a) Gaussian and (b) Triangular.



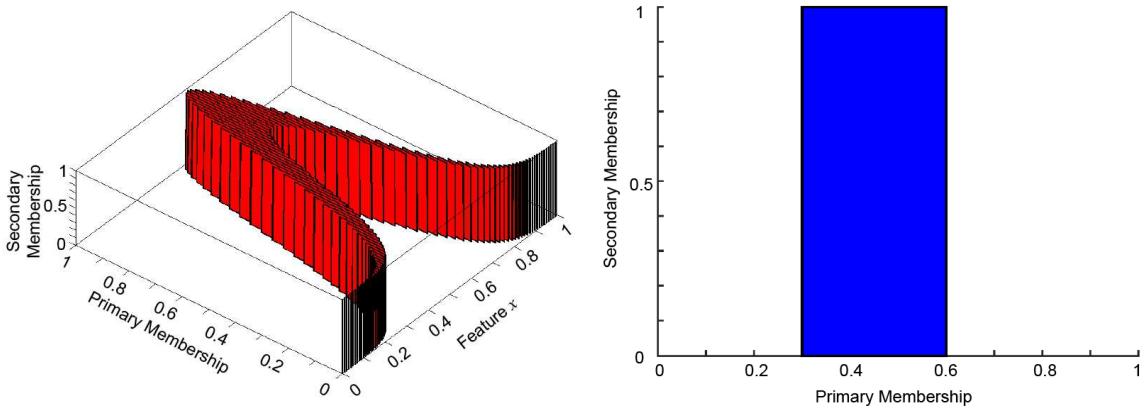
(a) (b)

Figure 4.2: T2 fuzzy sets (a) Gaussian fitting on a histogram for obtaining the upper membership function (UMF) and lower membership function (LMF) and (b) Vertical slice on UMF and LMF.



(a) (b)

Figure 4.3: T2 fuzzy membership function (a) the fuzzy membership function and (b) cross-section of the curve intersected by plane P at $x = x'$.



(a) (b)

Figure 4.4: IT2 fuzzy membership function (a) the fuzzy membership function and (b) cross-section of the curve intersected by plane P at $x = x'$.

reduces to T1 fuzzy set.

Definition 4.2 *For each value of x, say $x = x'$, the 2-D plane (Fig. 4.2(b), 4.3(b)) having axes u and $\mu_{\widetilde{A}}(x', u)$, is called the vertical slice [347] of $\mu_{\widetilde{A}}(x, u)$. A vertical slice of $\mu_{\widetilde{A}}(x, u)$ is actually a secondary membership function, represented as $\mu_{\widetilde{A}}(x = x', u)$ for $x' \in \mathbb{X}$ and $u \in [0, 1]$,*

$$\mu_{\widetilde{A}}(x = x', u) = \mu_{\widetilde{A}}(x') \equiv \int_{u \in J_{x'}} f_{x'}(u)/u, J_{x'} \subseteq [0, 1]; \tag{4.4}$$

where $0 \leq f_{x'}(u) \leq 1$. As $x' \in \mathbb{X}$, $\mu_{\widetilde{A}}(x)$ is referred as secondary membership function, which is a T1 fuzzy set and often referred as a secondary set. The amplitude of a secondary membership function, $f_x(u)$ or $\mu_{\widetilde{A}}(x, u)$, is called a secondary grade.

Definition 4.3 *The domain of a secondary membership function is referred as the primary membership [347] of x.*

In (4.3), J_x is the primary membership of x.

Definition 4.4 *Uncertainty in the primary memberships of a T2 fuzzy set $\mathbb{X}$ is a bounded region, called the footprint of uncertainty (FOU, denoted as $F\mathbb{X}$ [347]), which is the union of all primary memberships, that is,*

$$F\mathbb{X} = \bigcup_{x \in \mathbb{X}} J_x. \tag{4.5}$$

Let $F\mathbb{X}(x) \equiv J_x, \forall x \in \mathbb{X}$. Hence, (4.2) can be re-expressed as

$$\widetilde{A} = \int \int_{(x,u) \in F\mathbb{X}} \mu_{\widetilde{A}}(x, u)/(x, u). \tag{4.6}$$

The FOU represents the blurring of a T1 membership function.

Definition 4.5 *For any $\mathbb{X}$, there exist two T1 membership functions, namely, lower and upper membership functions, which create the bounds for $F\mathbb{X}$. The lower membership function (LMF) creates the lower bound of $F\mathbb{X}$, and the upper membership function (UMF) creates the upper bound of $F\mathbb{X}$ [347].*

$$\underline{\mu}_{\widetilde{A}}(x) = \{\underline{F\mathbb{X}}, \forall x \in \mathbb{X}\} = \inf\{u | u \in [0,1], \mu_{\widetilde{A}}(x,u) > 0\}; \tag{4.7}$$

$$\overline{\mu}_{\widetilde{A}}(x) = \{\overline{F\mathbb{X}}, \forall x \in \mathbb{X}\} = \sup\{u | u \in [0,1], \mu_{\widetilde{A}}(x,u) > 0\}. \tag{4.8}$$

So, the FOU can be represented as

$$F\mathbb{X} = \{(x,u) | x \in \mathbb{X}, u \in [\underline{\mu}_{\widetilde{A}}(x), \overline{\mu}_{\widetilde{A}}(x)]\}; \tag{4.9}$$

where $\underline{\mu}_{\widetilde{A}}(x)$ and $\overline{\mu}_{\widetilde{A}}(x)$ are, respectively, the lower membership function and upper membership function of x.

4.2.2 IT2 Fuzzy Sets

An implementation of general T2 fuzzy sets is more computationally intensive as compared to T1 fuzzy sets as it has secondary grades associated with each primary membership functions. The interval type-2 (IT2) fuzzy set is a special case of general T2 fuzzy set [283], where the third-dimension value, that is, all secondary grades for an IT2 fuzzy set are uniformly weighted (for example, $\mu_{\widetilde{A}}(x,u) = 1$) everywhere as shown in Fig. 4.3(a), that means, no new information is contained in the third dimension of an IT2 fuzzy set. So, the computational complexity is significantly reduced. In this case, the third dimension is not considered, and only the FOU is sufficient to describe the set.

Definition 4.6 *Let $\mathbb{X}$ be the universe of discourse for the primary variable x, $\mathbb{U}$ be the universe of discourse for the secondary variable u, both of them are nonempty finite sets and assumed to be $[0,1]$. An IT2 fuzzy set, denoted as $\widehat{A}$ (Fig.4.4), is characterized by an IT2 membership function $\mu_{\widehat{A}}(x,u)$, where $x \in \mathbb{X}$, $u \in [0,1]$ and $\mu_{\widehat{A}}(x,u) = 1$, that is,*

$$\widehat{A} = \{((x,u), \mu_{\widehat{A}}(x,u)) | x \in \mathbb{X}, u \in [0,1], \mu_{\widehat{A}}(x,u) = 1\}. \tag{4.10}$$

4.2.3 IT2 Fuzzy-Rough Sets

Let $< \mathbb{U}, \tilde{\mathbb{A}} >$ be an IT2 fuzzy approximation space, where $\mathbb{U} = \{x_1, \ldots, x_i, \ldots, x_n\}$ is the universe of discourse and $\tilde{\mathbb{A}}$ is the family of attributes. Considering $\mathbb{P}$ as a subset of attribute set $\tilde{\mathbb{A}}$, $\mathbb{U}/\mathbb{P} = \{\mathbb{F}_1, \ldots, \mathbb{F}_k\}$ be the IT2 fuzzy partition of $\mathbb{U}$, and $\mathbb{X} \in \mathbb{F}_{IT2}(\mathbb{U})$ be

an IT2 fuzzy concept to be approximated, where $\mathbb{F}_i$ is the ith IT2 fuzzy equivalence class. The IT2 fuzzy $\mathbb{P}$-lower and $\mathbb{P}$-upper approximations are defined as

$$\mu_{\underline{\mathbb{P}}\mathbb{X}(\mathbb{F}_i)} \equiv \int_{u \in F\underline{\mathbb{P}}\mathbb{X}(\mathbb{F}_i)} 1/u; \quad \mu_{\overline{\mathbb{P}}\mathbb{X}(\mathbb{F}_i)} \equiv \int_{u \in F\overline{\mathbb{P}}\mathbb{X}(\mathbb{F}_i)} 1/u; \quad \forall i = 1, 2, \ldots, k. \tag{4.11}$$

$$\text{where } F\underline{\mathbb{P}}\mathbb{X}(\mathbb{F}_i) \equiv \Big[\inf_{x \in \mathbb{U}} \max\{1 - \mu_{\underline{F}\mathbb{F}_i}(x), \mu_{\underline{F}\mathbb{X}}(x)\}, \inf_{x \in \mathbb{U}} \max\{1 - \mu_{\underline{F}\mathbb{F}_i}(x), \mu_{\overline{F}\mathbb{X}}(x)\} \Big];$$

$$\text{and } F\overline{\mathbb{P}}\mathbb{X}(\mathbb{F}_i) \equiv \Big[\sup_{x \in \mathbb{U}} \min\{1 - \mu_{\overline{F}\mathbb{F}_i}(x), \mu_{\underline{F}\mathbb{X}}(x)\}, \sup_{x \in \mathbb{U}} \min\{1 - \mu_{\overline{F}\mathbb{F}_i}(x), \mu_{\overline{F}\mathbb{X}}(x)\} \Big]. \tag{4.12}$$

Hence, the IT2 fuzzy lower and upper approximations can be redefined as

$$\mu_{\underline{\mathbb{P}}\mathbb{X}(x)} \equiv \int_{u \in F\underline{\mathbb{P}}\mathbb{X}(x)} 1/u; \quad \mu_{\overline{\mathbb{P}}\mathbb{X}(x)} \equiv \int_{u \in F\overline{\mathbb{P}}\mathbb{X}(x)} 1/u; \quad \forall i = 1, 2, \ldots, k. \tag{4.13}$$

$$\begin{aligned} \text{where } F\underline{\mathbb{P}}\mathbb{X}(x) \equiv \Big[& \sup_{\mathbb{F} \in \mathbb{U}/\mathbb{P}} \min\{\mu_{\underline{F}\mathbb{F}_i}(x), \inf_{y \in \mathbb{U}} \max\{1 - \mu_{\underline{F}\mathbb{F}_i}(y), \mu_{\underline{F}\mathbb{X}}(y)\}\}, \\ & \sup_{\mathbb{F} \in \mathbb{U}/\mathbb{P}} \min\{\mu_{\underline{F}\mathbb{F}_i}(x), \inf_{y \in \mathbb{U}} \max\{1 - \mu_{\underline{F}\mathbb{F}_i}(y), \mu_{\overline{F}\mathbb{X}}(y)\}\} \Big], \end{aligned} \tag{4.14}$$

$$\begin{aligned} \text{and } F\overline{\mathbb{P}}\mathbb{X}(x) \equiv \Big[& \sup_{\mathbb{F} \in \mathbb{U}/\mathbb{P}} \min\{\mu_{\overline{F}\mathbb{F}_i}(x), \sup_{y \in \mathbb{U}} \min\{1 - \mu_{\overline{F}\mathbb{F}_i}(y), \mu_{\underline{F}\mathbb{X}}(y)\}, \\ & \sup_{\mathbb{F} \in \mathbb{U}/\mathbb{P}} \min\{\mu_{\overline{F}\mathbb{F}_i}(x), \sup_{y \in \mathbb{U}} \min\{1 - \mu_{\overline{F}\mathbb{F}_i}(y), \mu_{\overline{F}\mathbb{X}}(y)\}\} \Big]. \end{aligned} \tag{4.15}$$

The membership of an object $x \in \mathbb{U}$ that belongs to the IT2 fuzzy positive region is defined by

$$\mu_{POS_{\mathbb{C}}(\mathbb{D})}(x) \equiv \sup_{X \in \mathbb{U}/\mathbb{D}} \mu_{\underline{\mathbb{C}}X}(x); \tag{4.16}$$

where $\mathbb{C}$ and $\mathbb{D}$ are condition and decision attribute sets, respectively. The lower and upper memberships of $\mu_{POS_{\mathbb{C}}(\mathbb{D})}(x)$ are denoted as $\underline{\mu}_{POS_{\mathbb{C}}(\mathbb{D})}(x)$ and $\overline{\mu}_{POS_{\mathbb{C}}(\mathbb{D})}(x)$, respectively, and corresponding IT2 fuzzy-rough dependency functions are

$$\underline{\gamma}_{\mathbb{C}}(\mathbb{D}) \equiv \frac{|\underline{\mu}_{POS_{\mathbb{C}}(\mathbb{D})}(x)|}{|\mathbb{U}|}, \quad \overline{\gamma}_{\mathbb{C}}(\mathbb{D}) \equiv \frac{|\overline{\mu}_{POS_{\mathbb{C}}(\mathbb{D})}(x)|}{|\mathbb{U}|}. \tag{4.17}$$

4.3 IT2 Fuzzy-Rough Attribute Selection Method

This section presents a feature selection algorithm, integrating judiciously the theory of IT2 fuzzy-rough sets and merit of maximal relevance-maximal significance (MRMS) criterion.

4.3.1 IT2 Fuzzy-Rough MRMS Method

The real life high dimensional data set may contain a number of irrelevant and insignificant features. The presence of such features may lead to a reduction in the useful information. Accordingly, a measure is required that can assess the effectiveness of a feature set. In this chapter, the theory of IT2 fuzzy-rough sets is used to select relevant and significant features from high dimensional data sets.

Let $\mathbb{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_i, \ldots, \mathcal{A}_j, \ldots, \mathcal{A}_m\}$ be the set of m condition attributes or features of a given data set and $\mathbb{S} \subseteq \mathbb{C}$ with cardinality $d < m$ is the set of selected features. Define $\underline{\gamma}_{\mathcal{A}_i}(\mathbb{D})$ and $\overline{\gamma}_{\mathcal{A}_i}(\mathbb{D})$ as the lower and upper relevance of the feature $\mathcal{A}_i$ with respect to the class labels $\mathbb{D}$, while $\underline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_i)$ and $\overline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_i)$ as the lower and upper significance of the feature $\mathcal{A}_i$ with respect to the set $\{\mathcal{A}_i, \mathcal{A}_j\}$. So the average upper and lower relevance of all selected features can be found by extending the (3.1) and (3.2) (reported in Chapter 3 for T1 fuzzy-rough sets) to IT2 fuzzy-rough sets as

$$\overline{\mathrm{R}} = \frac{1}{|\mathbb{S}|} \sum_{\mathcal{A}_i \in \mathbb{S}} \overline{\gamma}_{\mathcal{A}_i}(\mathbb{D}); \quad \underline{\mathrm{R}} = \frac{1}{|\mathbb{S}|} \sum_{\mathcal{A}_i \in \mathbb{S}} \underline{\gamma}_{\mathcal{A}_i}(\mathbb{D}); \tag{4.18}$$

while the average upper and lower significance among the selected features are

$$\overline{\mathrm{S}} = \frac{\sum\limits_{\mathcal{A}_i \neq \mathcal{A}_j \in \mathbb{S}} \{\overline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_i) + \overline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_j)\}}{|\mathbb{S}|(|\mathbb{S}| - 1)}, \tag{4.19}$$

$$\underline{\mathrm{S}} = \frac{\sum\limits_{\mathcal{A}_i \neq \mathcal{A}_j \in \mathbb{S}} \{\underline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_i) + \underline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}, \mathcal{A}_j)\}}{|\mathbb{S}|(|\mathbb{S}| - 1)}, \tag{4.20}$$

$$\text{that is, } \overline{\mathrm{S}} = \frac{\sum\limits_{\mathcal{A}_i \neq \mathcal{A}_j \in \mathbb{S}} 2\overline{\gamma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}) - \left\{\overline{\gamma}_{\mathcal{A}_i}(\mathbb{D}) + \overline{\gamma}_{\mathcal{A}_j}(\mathbb{D})\right\}}{|\mathbb{S}|(|\mathbb{S}| - 1)}, \tag{4.21}$$

$$\text{and } \underline{\mathrm{S}} = \frac{\sum\limits_{\mathcal{A}_i \neq \mathcal{A}_j \in \mathbb{S}} 2\underline{\gamma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}) - \left\{\underline{\gamma}_{\mathcal{A}_i}(\mathbb{D}) + \underline{\gamma}_{\mathcal{A}_j}(\mathbb{D})\right\}}{|\mathbb{S}|(|\mathbb{S}| - 1)}. \tag{4.22}$$

Therefore, the problem of selecting a set $\mathbb{S}$ of d relevant and significant features from the whole set $\mathbb{C}$ of m features is equivalent to optimize $\overline{\mathrm{R}}$ (respectively, $\underline{\mathrm{R}}$) and $\overline{\mathrm{S}}$ (respectively, $\underline{\mathrm{S}}$) simultaneously:

$$\max \Phi(\mathrm{R}, \mathrm{S}), \qquad \Phi = \mathrm{R} + \mathrm{S}; \tag{4.23}$$

where R and S, respectively, represent either $\underline{\mathrm{R}}$ and $\underline{\mathrm{S}}$ or $\overline{\mathrm{R}}$ and $\overline{\mathrm{S}}$, and operator $\Phi(\mathrm{R}, \mathrm{S})$ is defined to combine R and S.

4.3.2 Computation of Relevance and Significance

The relevance and significance of a feature, in the proposed IT2 fuzzy-rough MRMS method, are calculated based on the theory of IT2 fuzzy-rough sets. To compute both relevance and significance, the concepts of lower and upper fuzzy equivalence partition matrices are introduced next.

Given a finite set $\mathbb{U}$, $\mathbb{C}$ is an IT2 fuzzy attribute set in $\mathbb{U}$, which generates an IT2 fuzzy equivalence partition on $\mathbb{U}$. If c denotes the number of IT2 fuzzy equivalence classes generated by the IT2 fuzzy equivalence relation and n is the number of objects in $\mathbb{U}$, then c-partitions of $\mathbb{U}$ can be arrayed as two $(c \times n)$ matrices, namely, $\underline{\mathbb{M}}_{\mathbb{C}}$ and $\overline{\mathbb{M}}_{\mathbb{C}}$, termed as lower fuzzy equivalence partition matrix (LFEPM) and upper fuzzy equivalence partition matrix (UFEPM), respectively, and denoted by

$$\underline{\mathbb{M}}_{\mathbb{C}} = \begin{pmatrix} \underline{m}_{11}^{\mathbb{C}} & \dots & \underline{m}_{1j}^{\mathbb{C}} & \dots & \underline{m}_{1n}^{\mathbb{C}} \\ \dots & \dots & \dots & \dots & \dots \\ \underline{m}_{i1}^{\mathbb{C}} & \dots & \underline{m}_{ij}^{\mathbb{C}} & \dots & \underline{m}_{in}^{\mathbb{C}} \\ \dots & \dots & \dots & \dots & \dots \\ \underline{m}_{c1}^{\mathbb{C}} & \dots & \underline{m}_{c2}^{\mathbb{C}} & \dots & \underline{m}_{cn}^{\mathbb{C}} \end{pmatrix}; \tag{4.24}$$

$$\overline{\mathbb{M}}_{\mathbb{C}} = \begin{pmatrix} \overline{m}_{11}^{\mathbb{C}} & \dots & \overline{m}_{1j}^{\mathbb{C}} & \dots & \overline{m}_{1n}^{\mathbb{C}} \\ \dots & \dots & \dots & \dots & \dots \\ \overline{m}_{i1}^{\mathbb{C}} & \dots & \overline{m}_{ij}^{\mathbb{C}} & \dots & \overline{m}_{in}^{\mathbb{C}} \\ \dots & \dots & \dots & \dots & \dots \\ \overline{m}_{c1}^{\mathbb{C}} & \dots & \overline{m}_{c2}^{\mathbb{C}} & \dots & \overline{m}_{cn}^{\mathbb{C}} \end{pmatrix}; \tag{4.25}$$

where $\underline{m}_{ij}^{\mathbb{C}} \in [0, 1]$ and $\overline{m}_{ij}^{\mathbb{C}} \in [0, 1]$ represent, respectively, the lower and upper memberships of object x_j in the ith IT2 fuzzy equivalence class $\mathbb{F}_i$. These equations are basically the type-2 representation of the equations reported in Section 3.2.2 of Chapter 3.

Using the concepts of the LFEPM and UFEPM, the upper and lower relevance of the condition attribute $\mathcal{A}_i$ with respect to the decision attribute set $\mathbb{D}$ can be defined as

follows:

$$\underline{\gamma}_{\mathcal{A}_i}(\mathbb{D}) = \frac{1}{n}\sum_{j=1}^{n}\underline{\kappa}_j; \qquad 0 \leq \underline{\gamma}_{\mathcal{A}_i}(\mathbb{D}) \leq 1;$$

$$\text{where } \underline{\kappa}_j = \sup_k \sum_{u=\{\sup_s\{\min\{\underline{m}_{sj}^{\mathcal{A}_i},\ \inf_l\{\max\{1-\underline{m}_{sl}^{\mathcal{A}_i},\ \underline{m}_{kl}^{\mathbb{D}}\}\}\}\}\}}^{\{\sup_s\{\min\{\underline{m}_{sj}^{\mathcal{A}_i},\ \inf_l\{\max\{1-\underline{m}_{sl}^{\mathcal{A}_i},\ \overline{m}_{kl}^{\mathbb{D}}\}\}\}\}\}} 1/u; \tag{4.26}$$

$$\text{and } \overline{\gamma}_{\mathcal{A}_i}(\mathbb{D}) = \frac{1}{n}\sum_{j=1}^{n}\overline{\kappa}_j; \qquad 0 \leq \overline{\gamma}_{\mathcal{A}_i}(\mathbb{D}) \leq 1;$$

$$\text{where } \overline{\kappa}_j = \sup_k \sum_{u=\{\sup_s\{\min\{\overline{m}_{sj}^{\mathcal{A}_i},\ \inf_l\{\max\{1-\overline{m}_{sl}^{\mathcal{A}_i},\ \underline{m}_{kl}^{\mathbb{D}}\}\}\}\}\}}^{\{\sup_s\{\min\{\overline{m}_{sj}^{\mathcal{A}_i},\ \inf_l\{\max\{1-\overline{m}_{sl}^{\mathcal{A}_i},\ \overline{m}_{kl}^{\mathbb{D}}\}\}\}\}\}} 1/u. \tag{4.27}$$

The family of IT2 fuzzy sets produced by an IT2 fuzzy partitioning of the universe of discourse can play the role of IT2 fuzzy equivalence classes. The generalized π function in the one dimensional form can be used to assign membership values to different IT2 fuzzy equivalence classes for the input features. An IT2 fuzzy set with lower and upper membership functions $\underline{\pi}(x;\bar{c},\sigma)$ and $\overline{\pi}(x;\bar{c},\sigma)$, respectively, represents a set of points clustered around $\bar{c}$, where

$$\underline{\pi}(x;\bar{c},\sigma) = \begin{cases} 2^{p_l-1}\left(1-\frac{||x-\bar{c}||}{\sigma}\right)^{p_l} & \text{for } \frac{\sigma}{2} \leq ||x-\bar{c}|| \leq \sigma \\ 1-2^{p_h-1}\left(\frac{||x-\bar{c}||}{\sigma}\right)^{p_h} & \text{for } 0 \leq ||x-\bar{c}|| \leq \frac{\sigma}{2} \\ 0 & \text{otherwise} \end{cases}$$

$$\overline{\pi}(x;\bar{c},\sigma) = \begin{cases} 2^{p_h-1}\left(1-\frac{||x-\bar{c}||}{\sigma}\right)^{p_h} & \text{for } \frac{\sigma}{2} \leq ||x-\bar{c}|| \leq \sigma \\ 1-2^{p_l-1}\left(\frac{||x-\bar{c}||}{\sigma}\right)^{p_l} & \text{for } 0 \leq ||x-\bar{c}|| \leq \frac{\sigma}{2} \\ 0 & \text{otherwise} \end{cases}$$

where $p_h \geq p_l$, $\sigma > 0$ is the radius of the π functions with $\bar{c}$ as the central point and $||\cdot||$ denotes the Euclidean norm. The membership depends on the selection of fuzzifiers p_l and p_h. When the pattern x lies at the central point $\bar{c}$ of a class, then $||x-\bar{c}|| = 0$ and its membership value is maximum, that is, $\underline{\pi}(\bar{c};\bar{c},\sigma) = 1$ and $\overline{\pi}(\bar{c};\bar{c},\sigma) = 1$. The membership value of a point decreases as its distance from the central point $\bar{c}$, that is, $||x-\bar{c}||$ increases.

When $||x - \bar{c}|| = \left(\frac{\sigma}{2}\right)$, the membership value of x is 0.5 and this is called a crossover point. The fuzzifiers $p_l, p_h \in [p_{\min}, p_{\max}]$ can be varied to get the lower and upper membership functions for computing $\underline{\gamma}_{\mathcal{A}_i}(\mathbb{D})$ and $\overline{\gamma}_{\mathcal{A}_i}(\mathbb{D})$ according to (4.26) and (4.27), respectively, corresponding to the condition attribute $\mathcal{A}_i$.

The $c \times n$ LFEPM $\underline{\mathbb{M}}_{\mathcal{A}_i}$ and UFEPM $\overline{\mathbb{M}}_{\mathcal{A}_i}$, corresponding to ith feature $\mathcal{A}_i$, can be calculated from c-IT2 fuzzy equivalence classes of objects $x = \{x_1, \ldots, x_j, \ldots, x_n\}$, where

$$\underline{m}_{kj}^{\mathcal{A}_i} = \underline{\pi}(x_j; \bar{c}_k, \sigma_k); \quad \overline{m}_{kj}^{\mathcal{A}_i} = \overline{\pi}(x_j; \bar{c}_k, \sigma_k). \tag{4.28}$$

In effect, each position $\underline{m}_{kj}^{\mathcal{A}_i}$ (respectively, $\overline{m}_{kj}^{\mathcal{A}_i}$) of the LFEPM (respectively, UFEPM) $\underline{\mathbb{M}}_{\mathcal{A}_i}$ (respectively, $\overline{\mathbb{M}}_{\mathcal{A}_i}$) must satisfy the following conditions. For any value of k, if

$$\begin{aligned}
&\text{i) } s = \arg\max_j \{\underline{m}_{kj}^{\mathcal{A}_i}\}, \text{then } \max_j \{\underline{m}_{kj}^{\mathcal{A}_i}\} = \max_l \{\underline{m}_{ls}^{\mathcal{A}_i}\} > 0; \\
&\text{ii) } s = \arg\max_j \{\overline{m}_{kj}^{\mathcal{A}_i}\}, \text{then } \max_j \{\overline{m}_{kj}^{\mathcal{A}_i}\} = \max_l \{\overline{m}_{ls}^{\mathcal{A}_i}\} > 0.
\end{aligned}$$

The above axioms should hold for every fuzzy equivalence partition, which correspond to the requirement that an equivalence class is nonempty. Obviously, this definition degenerates to the normal definition of equivalence class when the equivalence relation is crisp. Similar to Section 3.2.2 of Chapter 3, in the present work, three IT2 fuzzy equivalence classes, namely, low, medium, and high, are considered.

To calculate the lower and upper significance of a condition attribute, the joint lower relevance $\underline{\gamma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D})$ and joint upper relevance $\overline{\gamma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D})$ between two attributes $\mathcal{A}_i$ and $\mathcal{A}_j$ need to be computed, which are the type-2 representation of the joint relevances reported in Section 3.2.2 of Chapter 3. The construction of resultant LFEPM $\underline{\mathbb{M}}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ and resultant UFEPM $\overline{\mathbb{M}}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ are necessary for computing the joint relevance. Let c_i and c_j be the number of IT2 fuzzy equivalence classes generated by the condition attributes $\mathcal{A}_i$ and $\mathcal{A}_j$, respectively. If r is the number of resultant IT2 fuzzy equivalence classes, then the $r \times n$ LFEPM $\underline{\mathbb{M}}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ and UFEPM $\overline{\mathbb{M}}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ can be computed as follows:

$$\underline{\mathbb{M}}_{\{\mathcal{A}_i,\mathcal{A}_j\}} = \underline{\mathbb{M}}_{\mathcal{A}_i} \cap \underline{\mathbb{M}}_{\mathcal{A}_j}; \qquad \overline{\mathbb{M}}_{\{\mathcal{A}_i,\mathcal{A}_j\}} = \overline{\mathbb{M}}_{\mathcal{A}_i} \cap \overline{\mathbb{M}}_{\mathcal{A}_j}; \tag{4.29}$$

where $\underline{m}_{kl}^{\{\mathcal{A}_i,\mathcal{A}_j\}} = \underline{m}_{pl}^{\mathcal{A}_i} \cap \underline{m}_{ql}^{\mathcal{A}_j}$, $\overline{m}_{kl}^{\{\mathcal{A}_i,\mathcal{A}_j\}} = \overline{m}_{pl}^{\mathcal{A}_i} \cap \overline{m}_{ql}^{\mathcal{A}_j}$, $k = (p-1)c_j + q$, and $\max\{c_i, c_j\} \leq r \leq c_i c_j$. In the present work, three IT2 fuzzy equivalence classes are considered, that is, $c_i = c_j = 3$.

The class membership function of decision attribute can be constructed as follows [394]: Suppose in a c-class problem, the vectors m_k and v_k represent the mean and standard deviation, respectively, for the kth class. The weighted distance of the training object x_i

from the kth class can be defined as

$$d_{ik} = \frac{x_i - m_k}{v_k}, \forall k = 1, \ldots, c. \tag{4.30}$$

As the feature with higher variance should have lesser weight, the weight $(1/v_k)$ is used in (4.30). If all feature values of any class are same, v_k can be considered as 1, that is, the feature is certainly an important feature and its contribution to the membership function should not be reduced. The membership of the ith pattern of the kth class is

$$\mu_k(x_i) = \frac{1}{1 + (\frac{d_{ik}}{c_d})^{c_e}}; \qquad \mu_k(x_i) \in [0, 1]; \tag{4.31}$$

where c_d and c_e are the denominational and exponential positive constants those control the amount of fuzziness. The parameters c_d and c_e can be varied in the range of $[c_{d_1}, c_{d_2}]$ and $[c_{e_1}, c_{e_2}]$, respectively, to get the lower and upper membership functions corresponding to the decision attribute. Hence, if an object x_i is present in the kth class, its lower membership value is $\mu_k(x_i)$, while its upper membership value is 1. On the other hand, if an object x_i is absent in the kth class, its lower membership value is 0 and upper membership value is $\mu_k(x_i)$.

4.3.3 IT2FR-MRMS Algorithm

The following greedy algorithm is used to solve (4.23) for selecting relevant and significant real valued features based on the theory of IT2 fuzzy-rough sets:

- *Input:* Original set $\mathbb{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_k, \ldots, \mathcal{A}_m\}$.
- *Output:* Reduced set $\mathbb{S}$.

1. Initialize $\mathbb{C} \leftarrow \{\mathcal{A}_1, \ldots, \mathcal{A}_i, \ldots, \mathcal{A}_j, \ldots, \mathcal{A}_m\}$, $\mathbb{S} \leftarrow \emptyset$.
2. Calculate the centers and radii of three π IT2 fuzzy sets for each feature $\mathcal{A}_i \in \mathbb{C}$ corresponding to $\underline{\pi}$ and $\overline{\pi}$.
3. Construct the LFEPM $\underline{\mathbb{M}}_{\mathcal{A}_i}$ and UFEPM $\overline{\mathbb{M}}_{\mathcal{A}_i}$ for each feature $\mathcal{A}_i \in \mathbb{C}$ according to (4.28).
4. Calculate the lower $\underline{\gamma}_{\mathcal{A}_i}(\mathbb{D})$ and upper $\overline{\gamma}_{\mathcal{A}_i}(\mathbb{D})$ relevance of each feature $\mathcal{A}_i \in \mathbb{C}$ according to (4.26) and (4.27).
5. Select the feature $\mathcal{A}_i$ as the most relevant feature that has the highest lower relevance value $\underline{\gamma}_{\mathcal{A}_i}(\mathbb{D})$. In effect, $\mathcal{A}_i \in \mathbb{S}$ and $\mathbb{C} = \mathbb{C} \setminus \mathcal{A}_i$.
6. Repeat following four steps until desired number of features $\underline{d}$ is selected or $\mathbb{C} = \emptyset$.

(a) Construct the resultant LFEPM $\underline{\mathbb{M}}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ of each of the remaining features $\mathcal{A}_j \in \mathbb{C}$ with respect to each selected feature $\mathcal{A}_i \in \mathbb{S}$ using (4.29).

(b) Calculate the lower significance of $\mathcal{A}_j \in \mathbb{C}$ with respect to each of the selected features $\mathcal{A}_i \in \mathbb{S}$ as follows:

$$\underline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D},\mathcal{A}_j) = \underline{\gamma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}) - \underline{\gamma}_{\mathcal{A}_i}(\mathbb{D}). \tag{4.32}$$

(c) Remove $\mathcal{A}_j$ from $\mathbb{C}$ if $\underline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D},\mathcal{A}_j) = 0$ for any feature $\mathcal{A}_i \in \mathbb{S}$.

(d) From the remaining features of $\mathbb{C}$, select feature $\mathcal{A}_j$ that maximizes the following conditions:

$$\omega\underline{\gamma}_{\mathcal{A}_j}(\mathbb{D}) + \frac{(1-\omega)}{|\mathbb{S}|}\sum_{\mathcal{A}_i\in\mathbb{S}} \underline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D},\mathcal{A}_j). \tag{4.33}$$

As a result of that, $\mathcal{A}_j \in \mathbb{S}$ and $\mathbb{C} = \mathbb{C} \setminus \mathcal{A}_j$.

7. Re-initialize $\mathbb{C} \leftarrow \{\mathcal{A}_1,\ldots,\mathcal{A}_i,\ldots,\mathcal{A}_j,\ldots,\mathcal{A}_m\}$, and then $\mathbb{C} = \mathbb{C} \setminus \mathbb{S}$.

8. Repeat following four steps until desired number of features $\overline{d} = d - \underline{d}$ is selected or $\mathbb{C} = \emptyset$.

(a) Construct the resultant UFEPM $\overline{\mathbb{M}}_{\{\mathcal{A}_i,\mathcal{A}_j\}}$ of each of the remaining features $\mathcal{A}_j \in \mathbb{C}$ with respect to each selected feature $\mathcal{A}_i \in \mathbb{S}$ using (4.29).

(b) Calculate the upper significance of $\mathcal{A}_j \in \mathbb{C}$ with respect to each of the selected features $\mathcal{A}_i \in \mathbb{S}$ as follows:

$$\overline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D},\mathcal{A}_j) = \overline{\gamma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D}) - \overline{\gamma}_{\mathcal{A}_i}(\mathbb{D}). \tag{4.34}$$

(c) Remove $\mathcal{A}_j$ from $\mathbb{C}$ if $\overline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D},\mathcal{A}_j) = 0$ for any feature $\mathcal{A}_i \in \mathbb{S}$.

(d) From the remaining features of $\mathbb{C}$, select feature $\mathcal{A}_j$ that maximizes the following conditions:

$$\omega\overline{\gamma}_{\mathcal{A}_j}(\mathbb{D}) + \frac{(1-\omega)}{|\mathbb{S}|}\sum_{\mathcal{A}_i\in\mathbb{S}} \overline{\sigma}_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathbb{D},\mathcal{A}_j). \tag{4.35}$$

As a result of that, $\mathcal{A}_j \in \mathbb{S}$ and $\mathbb{C} = \mathbb{C} \setminus \mathcal{A}_j$.

9. Stop.

4.3.4 Complexity of the Algorithm

The IT2FR-MRMS has low computational complexity with respect to the number of features and samples in original data set. The steps 2, 3, and 4 of the proposed IT2FR-MRMS

algorithm are executed m times for m attributes. The complexity to compute the centers and radii of three IT2 fuzzy sets for each attribute, which is carried out in step 2, is $\mathcal{O}(n)$. The construction of both LFEPM and UFEPM of each feature performed in step 3 has $\mathcal{O}(nc)$ time complexity. The computation of the relevance of each feature is carried out in step 4, which has $\mathcal{O}(nc\tilde{c})$ time complexity, where $\tilde{c}$ represents the number of IT2 fuzzy equivalence classes of decision attribute. Hence, the overall time complexity of steps 2, 3, and 4 for m features is $\mathcal{O}(mnc\tilde{c})$. The selection of most relevant feature from the set of m features, which is carried out in step 5, has a complexity $\mathcal{O}(m)$. There is only one loop in both of the steps 6 and 8 of the proposed feature selection method, which are executed $(\underline{d}-1)$ times and $\overline{d}$ times, respectively, where $d = \underline{d} + \overline{d}$ represents the number of selected features. The construction of the resultant LFEPM and UFEPM, which is carried out in steps 6.(a) and 8.(a), and the computation of lower and upper significance of a candidate feature with respect to an already-selected feature, which is carried out in steps 6.(b) and 8.(b), have $\mathcal{O}(nc^2)$ and $\mathcal{O}(nc^2\tilde{c})$ time complexity, respectively. If $\acute{m} < m$ represents the cardinality of the already-selected feature set, the total complexity of steps 6.(a), 6.(b), 8.(a) and 8.(b) is $\mathcal{O}((m-\acute{m})(nc^2\tilde{c}))$. The selection of a feature from $(m-\acute{m})$ candidate features by maximizing both relevance and significance, which is carried out in steps 6.(d) and 8.(d), has a complexity $\mathcal{O}(m-\acute{m})$. Hence, the total complexity to execute two loops $(\underline{d}-1)$ and $\overline{d}$ times is $\mathcal{O}((d-1)(m-\acute{m})(nc^2\tilde{c}))$.

In effect, the selection of a set of d relevant and significant features from the whole set of m features using the proposed IT2 fuzzy-rough set based first order incremental search method has an overall computational complexity of $\mathcal{O}(mndc^2\tilde{c})$.

4.4 Experimental Results and Discussion

The performance of the proposed IT2FR-MRMS is extensively studied and compared with that of different feature selection and extraction algorithms. The algorithms compared are mutual information based mRMR framework (classical mRMR) [416] and InfoGain [430]; rough set based quick reduct [83] and MRMS framework [327]; T1 fuzzy-rough set based quick reduct (fuzzy-rough quick reduct) [220], MRMS method (T1 fuzzy-rough MRMS) [315], and mRMR method (fuzzy-rough mRMR) [323]; IT2 fuzzy-rough quick reduct [531]; margin based approaches such as relevance in estimating features (RELIEF) [252] and iterative search margin based algorithm (SIMBA) [152]; and existing feature extraction algorithms, namely, principal component analysis (PCA), independent component analysis (ICA), and linear discriminant analysis (LDA) [112]. The performance of the IT2 fuzzy-rough sets and the MRMS criterion is also compared with that of other evaluation criteria, namely, Max-Relevance and Max-Dependency, several feature evaluation indices, namely, class separability index, Davies-Bouldin (DB) index, Dunn index,

and fuzzy feature evaluation (FFE) index, and various rough set models such as classical rough sets, neighborhood rough sets, and T1 fuzzy-rough sets.

The multiplicative parameter η in (3.11) of fuzzy-rough sets is set to 1.5, while the weight parameter ω in (4.33) and (4.35) and that of mRMR method are set to 0.5 [315]. The parameters for computation of $\underline{\pi}$ and $\overline{\pi}$, that is, p_l and p_h are varied taking values from the set {1.1, 1.5, 2.0, 2.5, 3.0} for all the values of $p_h \geq p_l$. The parameters for decision attribute, that is, c_d and c_e can take values from the sets {2, 3} and {2, 3, 4, 5}, respectively [394]. Three pattern classifiers, namely, K-nearest neighbor (K-NN) rule, support vector machine (SVM), and C4.5 decision tree, are used to evaluate the performance of different dimensionality reduction methods. To evaluate the performance of different dimensionality reduction methods, several benchmark data sets such as Satimage, Segmentation, Leukemia I, Leukemia II, Breast Cancer I, Breast Cancer II, Colon Cancer, Lung Cancer, Isolet, and Multiple Features are used. A brief description of these data sets and three pattern classifiers is reported in Section 3.3 of Chapter 3.

To compute the classification accuracy of the K-NN, SVM, and C4.5, the training-testing, 10-fold cross validation (CV), and bootstrap $\mathcal{E}_{\text{BS}}^{.632}$ approach are used. Both 10-fold CV and .632+ bootstrap approach ($\mathcal{E}_{\text{BS}}^{.632}$) are performed on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features data sets, while the training-testing is done on Satimage, Segmentation, Leukemia II, and Breast II data sets. The procedure to calculate $\mathcal{E}_{\text{BS}}^{.632}$ is reported in Appendix A. In this chapter, the .632+ bootstrap accuracy $(1 - \mathcal{E}_{\text{BS}}^{.632}) \times 100$ values are presented in Tables 4.7, 4.8, and 4.9 to make it analogous with the classification accuracy of training-testing reported in Table 3.1 of Chapter 3, Tables 4.2, and 4.3. The means and standard deviations of the classification accuracy of the K-NN, SVM and C4.5 are computed for 10-fold CV. Tests of significance, reported in Section 3.4.1 of Chapter 3, are performed for the inequality of means (of the classification accuracy of the K-NN, SVM and C4.5) obtained using the IT2FR-MRMS method and other approaches. Since both mean pairs and variance pairs are unknown and different, classical Behrens-Fisher hypothesis testing is used. Tables 4.4, 4.5, and 4.6 report the values of test statistic computed, where the individual means, number of features selected, and standard deviations are reported for IT2FR-MRMS. The corresponding tabled value is 1.81 at an error probability level of 0.05. If the computed value is greater than the tabled value, the means are significantly different.

4.4.1 Optimum Values of Different Parameters

The parameters p_l and p_h ($p_h \geq p_l$) control the shape of two membership functions, namely, lower membership function $\underline{\pi}$ and upper membership function $\overline{\pi}$. Keeping the values of other parameters fixed, the membership functions can be altered varying p_l and p_h. On the other hand, the parameters c_d and c_e in (4.31) control the amount of fuzziness of the

Table 4.1: Optimum Values of Parameters for IT2 Fuzzy-Rough Sets

Data Sets	$S^\star = \{p_l^\star, p_h^\star, c_d^\star, c_e^\star\}$	
	Training/Testing	
Satimage	{1.5, 3.0, 3, 5}	
Segmentation	{1.5, 2.5, 2/3, 2/3/4/5}	
Leukemia II	{2.5, 2.5, 3, 3/5}	
Breast Cancer II	{1.5, 2.0, 3, 4}	
	10-fold Cross Validation	Bootstrap $\mathcal{E}_{\text{BS}}^{632}$
Colon Cancer	{2.0, 2.0, 2, 3}/{2.0, 2.5, 3, 4}	{2.0, 2.5, 2, 3}
Breast Cancer I	{2.0, 2.5, 3, 3}	{2.0, 2.5, 3, 4}
Lung Cancer	{2.0, 2.5, 2, 3/4/5}	{2.0, 2.0, 2/3, 3/4}/{2.0, 2.5, 2, 4}
Leukemia I	{1.5, 2.0, 2/3, 3/4}	{1.5, 2.5, 3, 2/3/4}
Isolet	{2.0, 2.5, 2/3, 3/4}	{2.5, 2.5, 3, 2/3}
Multiple Features	{2.0, 2.5, 2, 3/4}	{2.0, 2.5, 2, 4}

membership of any sample in a particular decision class.

Let $S = \{p_l, p_h, c_d, c_e\}$ be the set of parameters and $S^* = \{p_l^\star, p_h^\star, c_d^\star, c_e^\star\}$ is the set of optimal parameters. To find out the optimum values of p_l, p_h, c_d, and c_e for a given data set, the class separability index [101] is used. The class separability index $\mathcal{S}$ of a data set is defined as $\mathcal{S} = \text{trace}(V_W^{-1}V_B)$, where V_W is the within class scatter matrix and V_B is the between class scatter matrix, defined as follows:

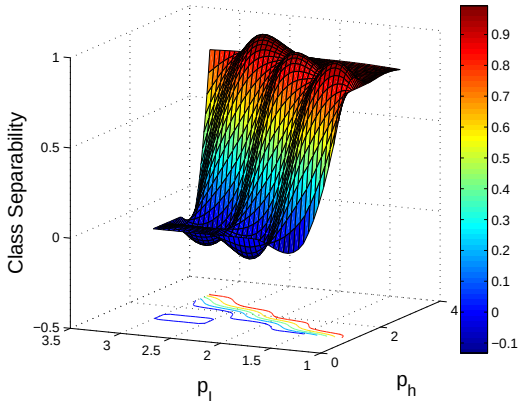
$$V_W = \sum_{j=1}^{c} \pi_j E\{(X-\mu_j)(X-\mu_j)^T | c_j\} = \sum_{j=1}^{c} \pi_j \Sigma_j; \tag{4.36}$$

$$V_B = \sum_{j=1}^{c} \pi_j (\mu_j - \bar{\mu})(\mu_j - \bar{\mu})^T; \text{ and} \tag{4.37}$$

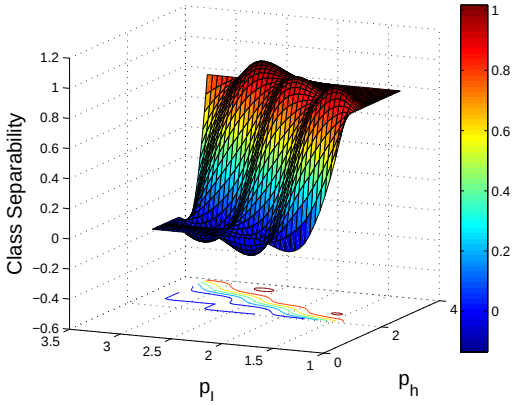
$$\bar{\mu} = E\{X\} = \sum_{j=1}^{c} \pi_j \mu_j; \tag{4.38}$$

where c is the number of classes, π_j is a priori probability that a pattern belongs to class c_j, X is a feature vector, $\bar{\mu}$ is the sample mean vector for the entire data points, μ_j and Σ_j represent sample mean and covariance matrix of class c_j, respectively and $E\{\cdot\}$ is the expectation operator. A higher value of $\mathcal{S}$ ensures that classes are well separated by their scatter means.

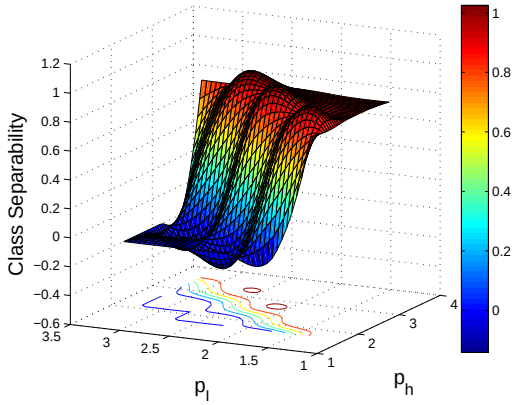
Fig. 4.5 and 4.6 present the variation of $\mathcal{S}$ index for the IT2FR-MRMS sets with respect to different values of p_l and p_h considering the optimum values of c_d and c_e, while Fig. 4.7 and 4.8 report that with respect to different values of c_d and c_e considering $p_l^\star$ and $p_h^\star$. From the results reported in Fig. 4.5, 4.6, 4.7, and 4.8, it is seen that as the values of p_l, p_h, c_d, and c_e increase, the $\mathcal{S}$ index also increases and attains its maximum value at the particular values of $p_l^\star$, $p_h^\star$, $c_d^\star$, and $c_e^\star$. Hence, the optimum values of $p_l^\star$, $p_h^\star$, $c_d^\star$, and $c_e^\star$ are



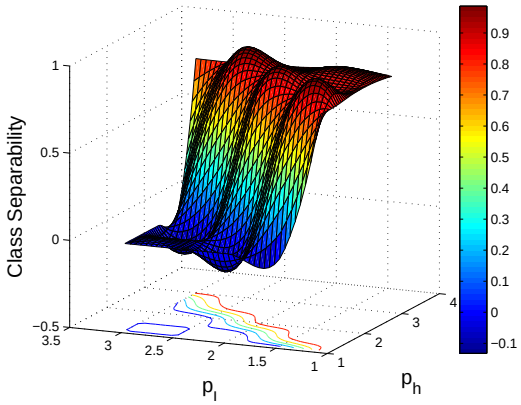
(a) Colon ($c_d^\star = 3$, $c_e^\star = 4$)



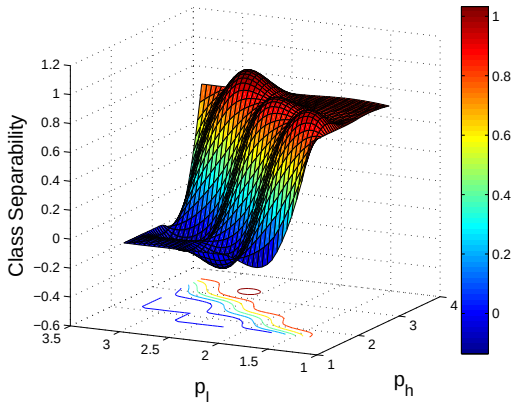
(b) Breast I ($c_d^\star = 3$, $c_e^\star = 3$)



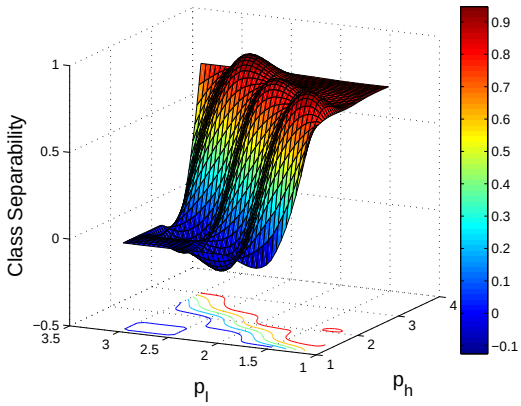
(c) Lung ($c_d^\star = 2$, $c_e^\star = 4$)



(d) Leukemia I ($c_d^\star = 2$, $c_e^\star = 3$)

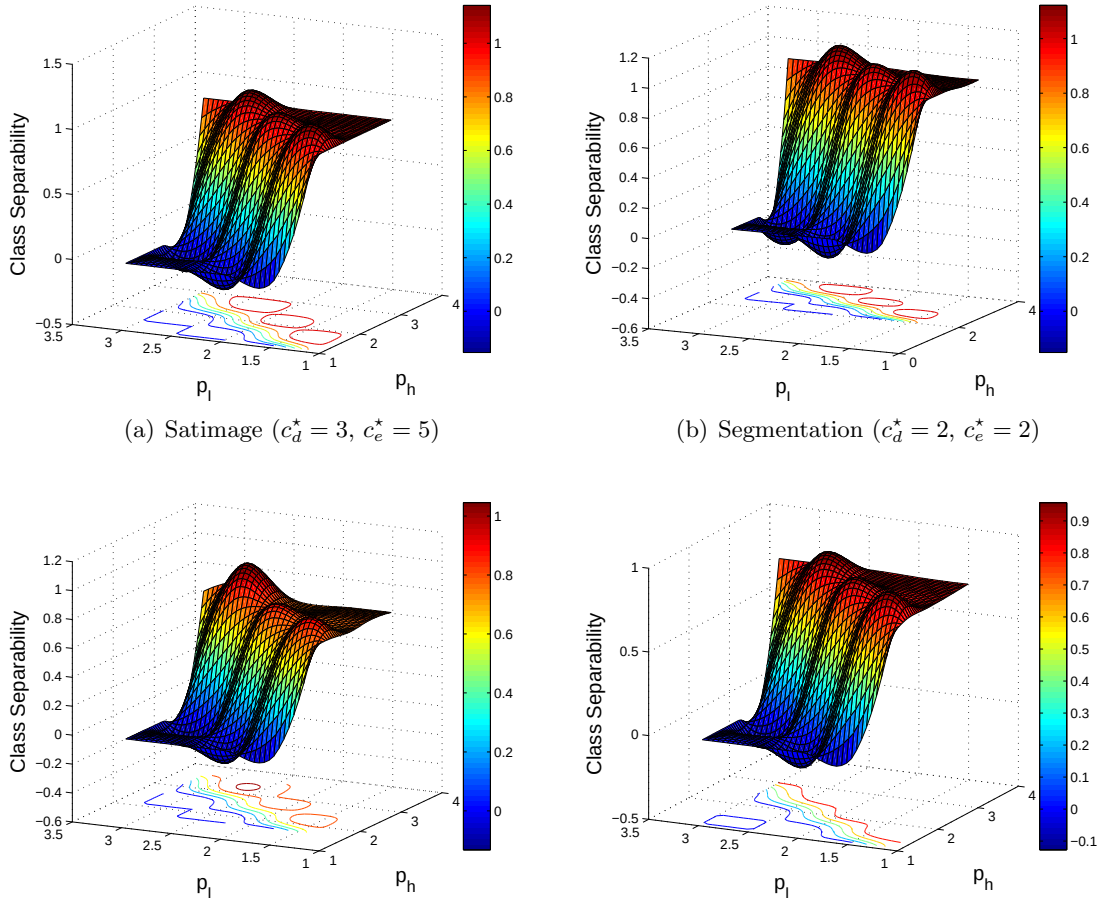


(e) Isolet ($c_d^\star = 2$, $c_e^\star = 3$)



(f) Multiple Features ($c_d^\star = 2$, $c_e^\star = 3$)

Figure 4.5: Variation of class separability index for different values of p_l and p_h



(a) Satimage ($c_d^\star = 3$, $c_e^\star = 5$)

(b) Segmentation ($c_d^\star = 2$, $c_e^\star = 2$)

(c) Leukemia II ($c_d^\star = 3$, $c_e^\star = 3$)

(d) Breast II ($c_d^\star = 3$, $c_e^\star = 4$)

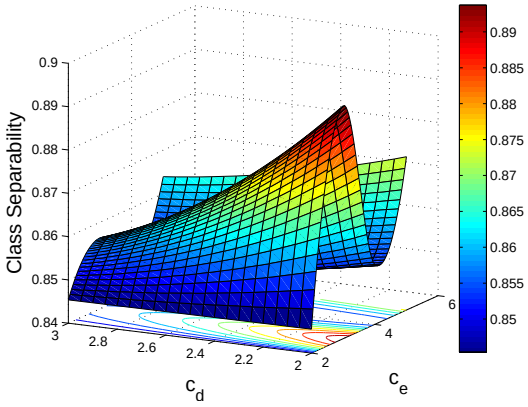
Figure 4.6: Variation of class separability index for different values of p_l and p_h

obtained using the following relation:

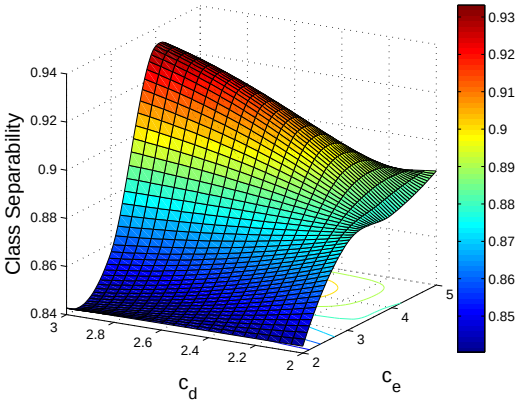
$$\mathrm{S}^\star = \arg\max_{\mathrm{S}} \{\mathcal{S}\}. \tag{4.39}$$

Table 4.1 presents the optimum values of p_l, p_h, c_d, and c_e for the IT2FR-MRMS obtained using (4.39) on different data sets. From the results reported in Table 4.1, it can be seen that the optimum values of $p_l^\star$ and $p_h^\star$ vary, respectively, in between 1.5 to 2.5 and 2.0 to 3.0, while that of $c_d^\star$ and $c_e^\star$ vary, respectively, in between 2 to 3 and 2 to 5.

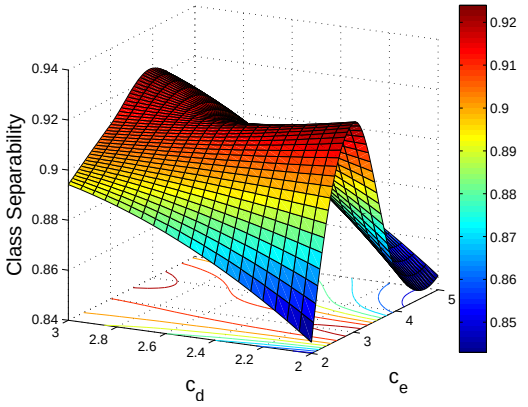
In order to establish the effectiveness of the proposed method for finding optimum values of different parameters, extensive experimentation is done on different real life data sets. Table 4.2 presents the performance of the IT2FR-MRMS on Satimage, Segmentation, Leukemia II, and Breast II data sets based on training-testing, while the performance on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features data sets based on both



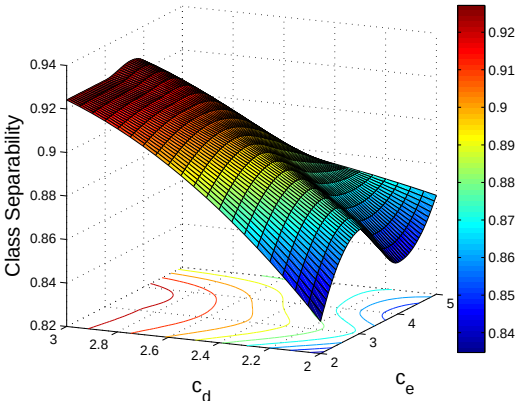
(a) Colon ($p_l^\star = 2.0$, $p_h^\star = 2.5$)



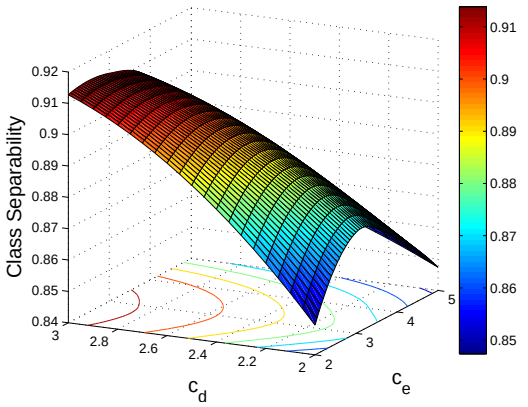
(b) Breast I ($p_l^\star = 2.0$, $p_h^\star = 2.5$)



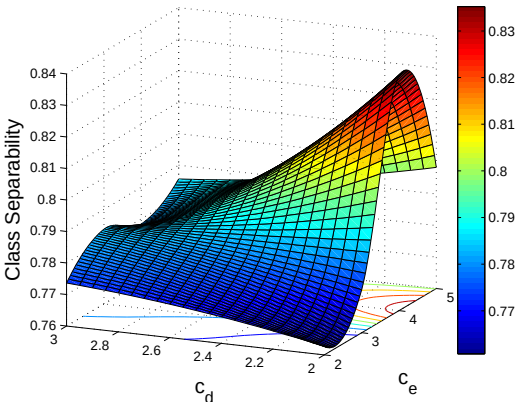
(c) Lung ($p_l^\star = 2.0$, $p_h^\star = 2.5$)



(d) Leukemia I ($p_l^\star = 1.5$, $p_h^\star = 2.0$)

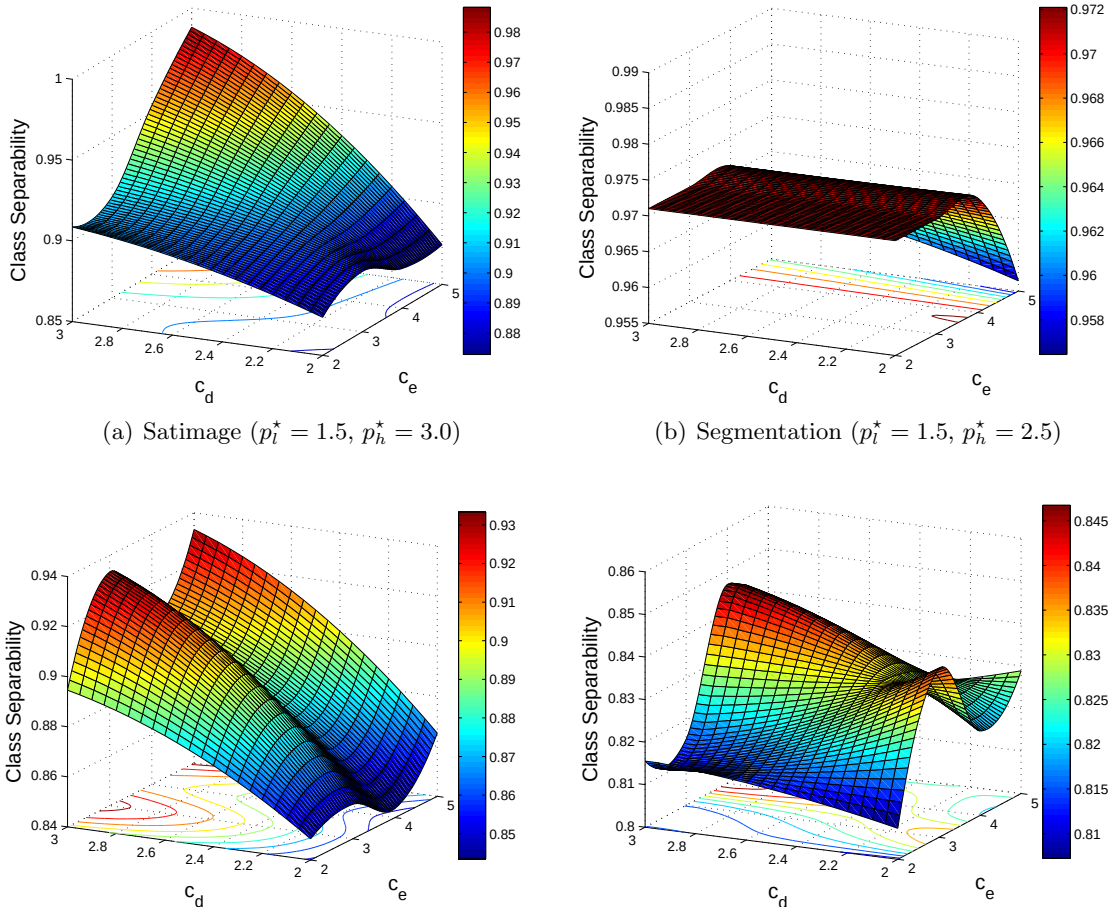


(e) Isolet ($p_l^\star = 2.0$, $p_h^\star = 2.5$)



(f) Multiple Features ($p_l^\star = 2.0$, $p_h^\star = 2.5$)

Figure 4.7: Variation of class separability index for different values of c_d and c_e



(a) Satimage ($p_l^\star = 1.5$, $p_h^\star = 3.0$)

(b) Segmentation ($p_l^\star = 1.5$, $p_h^\star = 2.5$)

(c) Leukemia II ($p_l^\star = 2.5$, $p_h^\star = 2.5$)

(d) Breast II ($p_l^\star = 1.5$, $p_h^\star = 2.0$)

Figure 4.8: Variation of class separability index for different values of c_d and c_e

10-fold CV and bootstrap $\mathcal{E}_{\text{BS}}^{.632}$ approach are also presented in this table. The results and subsequent discussions are analyzed in this table with respect to classification accuracy of the K-NN, SVM, and C4.5. The best test accuracy obtained from all possible parameter values on each data set is compared with the test accuracy corresponding to best training accuracy and that for optimum parameters. All the results reported in Table 4.2 confirm that the training–testing, bootstrap $\mathcal{E}_{\text{BS}}^{.632}$, and 10-fold CV accuracy obtained using optimum parameters are comparable or better than the accuracy corresponding to best training accuracy and the best test accuracy in most of the cases, irrespective of the classifiers and data sets used. Out of 48 cases, the accuracy obtained using the proposed technique is exactly same with the best test accuracy in 38 cases, whereas the accuracy corresponding to best training is better than that of the proposed technique in only 3 cases.

Table 4.2: Classification Accuracy for Optimum Parameters on Different Data Sets in case of IT2FR-MRMS

Experimental Setup	Different Data Sets	Test Accuracy of K-NN			Test Accuracy of SVM			Test Accuracy of C4.5		
		Best	Best Training	Proposed	Best	Best Training	Proposed	Best	Best Training	Proposed
Training / Testing	Satimage	86.6/12	86.2/11	**86.6/12**	86.2/25	86.2/25	**86.2/25**	86.9/25	86.9/25	**86.9/25**
	Segmentation	86.0/10	86.0/10	**86.0/10**	**92.4/11**	92.4/14	91.8/11	91.1/13	91.1/13	**91.1/16**
	Leukemia II	90.2/36	89.3/49	**90.2/36**	90.2/42	90.2/49	**90.2/42**	88.4/19	87.5/50	**88.4/19**
	Breast II	89.7/18	87.9/26	**89.7/21**	94.4/19	90.3/28	**94.4/19**	94.7/18	92.6/27	**94.7/18**
10-fold Cross Validation	Colon	74.4/13	73.2/11	**74.4/13**	71.4/28	71.4/30	**71.4/28**	83.0/37	80.2/26	**83.0/39**
	Breast I	73.7/12	72.3/10	**73.7/12**	**64.7/21**	63.2/15	64.4/20	83.2/12	81.4/17	**83.2/12**
	Lung	**73.9/12**	73.9/12	73.6/10	70.7/11	70.4/10	**70.7/11**	**85.0/24**	84.7/21	84.7/21
	Leukemia I	81.4/16	81.1/16	**81.4/16**	82.1/44	82.1/44	**82.1/44**	90.2/28	88.3/31	**90.2/28**
	Isolet	**73.4/36**	72.2/22	73.2/32	82.5/20	81.1/21	**82.5/24**	92.1/28	91.3/23	**92.1/28**
	Multiple Features	80.3/23	80.2/20	**80.3/23**	77.3/31	77.3/31	**77.3/31**	87.3/21	87.3/21	**87.3/21**
Bootstrap $\mathcal{E}_{\text{BS}}^{.632}$	Colon	85.1/15	84.3/14	**85.1/23**	71.2/23	71.1/38	**71.2/23**	81.1/34	81.1/34	**81.1/34**
	Breast I	**75.9/26**	73.8/16	75.7/12	76.6/31	76.2/37	**76.6/31**	82.6/21	82.1/34	**82.6/22**
	Lung	90.7/18	90.7/23	**90.7/20**	**81.0/43**	80.2/41	80.8/42	80.8/45	79.9/38	**80.8/45**
	Leukemia I	89.2/29	88.8/11	88.8/11	71.1/27	70.2/30	**71.1/28**	74.6/28	74.2/36	**74.6/30**
	Isolet	**82.9/8**	80.2/21	**82.9/8**	**82.3/47**	82.3/47	82.1/46	**77.6/35**	77.1/34	77.3/33
	Multiple Features	84.9/9	82.7/16	**84.9/8**	88.9/46	88.1/38	**88.9/46**	83.5/46	83.0/40	**83.5/46**

Table 4.3: Performance of IT2 Fuzzy-Rough sets on Satimage, Segmentation, Leukemia II, and Breast II

Different Criteria	Satimage			Segmentation			Leukemia II			Breast II		
	K-NN	SVM	C4.5	K-NN	SVM	C4.5	K-NN	SVM	C4.5	K-NN	SVM	C4.5
Max Relevance	86.1/21	86.0/33	86.8/15	86.0/12	91.5/14	91.1/13	83.9/42	90.2/49	83.0/45	83.6/35	82.4/23	82.1/31
Max Dependency	85.8/5	83.4/5	85.2/5	70.1/3	75.0/3	76.4/3	87.5/16	88.4/19	88.4/19	86.3/34	87.2/29	87.7/20
MRMS	**86.6/12**	**86.2/25**	**86.9/25**	**86.0/10**	**92.4/11**	**91.1/16**	**90.2/36**	**90.2/42**	88.4/19	**89.7/21**	94.4/19	**94.7/18**

Table 4.4: Statistical Significance Test of Different Rough Sets on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features using 10-fold CV

Different Classifiers	Different Criteria	Different Rough Sets	Different Data Sets					
			Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	Max Relevance	Classical	**3.077996**	**2.280527**	**2.847329**	**2.739347**	**5.232904**	**5.135190**
		Neighborhood	**2.954067**	1.306499	**1.879968**	1.521296	**2.574744**	**4.607906**
		T1 Fuzzy	1.706103	0.592883	1.291052	1.409944	1.383596	**2.694099**
		IT2 Fuzzy	1.780929	0.592883	0.941554	1.409944	1.295921	**2.743860**
	Max Dependency	Classical	1.739482	**2.280527**	1.520972	**1.847185**	**4.737911**	**4.893430**
		Neighborhood	0.826860	1.778155	0.913951	1.778055	0.756816	**3.672023**
		T1 Fuzzy	1.253397	1.508738	0.941554	1.472595	0.881812	**3.046778**
		IT2 Fuzzy	0.784971	0.856056	0.941554	1.690492	1.002359	**3.046778**
	MRMS	Classical	**3.381321**	**2.280527**	0.689997	1.311911	**3.289563**	1.387702
		Neighborhood	**2.119770**	1.801425	0.636911	1.091225	0.732637	0.887698
		T1 Fuzzy	0.013851	0.152056	0.039378	0.506719	0.000000	0.239565
		IT2 Fuzzy	(74.35/13, 11.12)	(71.40/28, 14.25)	(82.98/39, 8.22)	(81.44/16, 11.32)	(82.12/44, 5.83)	(90.21/28, 4.32)
SVM	Max Relevance	Classical	**2.078169**	1.608815	**2.902289**	0.962639	**5.042815**	**3.648460**
		Neighborhood	1.654570	1.334807	**1.991845**	0.952360	**1.985572**	**3.490680**
		T1 Fuzzy	0.674894	0.795882	1.713939	0.985817	1.407685	**3.001879**
		IT2 Fuzzy	0.934268	0.794087	1.713939	0.985817	1.375413	**1.854591**
	Max Dependency	Classical	**1.920221**	1.654741	1.558364	0.926554	**5.226582**	**4.545109**
		Neighborhood	1.570417	1.528257	1.539048	1.348550	1.354192	**3.093808**
		T1 Fuzzy	1.600464	0.935289	1.223868	1.411831	1.320932	**2.190916**
		IT2 Fuzzy	1.530655	0.935289	0.816187	1.300437	1.220906	**2.190916**
	MRMS	Classical	1.387248	1.500112	1.208107	0.926554	**3.196621**	1.349149
		Neighborhood	0.551271	0.847889	0.846835	0.863450	1.405368	0.847053
		T1 Fuzzy	0.017597	0.180226	0.127194	0.481437	0.045773	0.000000
		IT2 Fuzzy	(73.68/12, 13.69)	(64.40/20, 10.23)	(83.23/12, 6.33)	(73.21/32, 13.00)	(82.45/24, 5.64)	(92.10/28, 3.34)
C4.5	Max Relevance	Classical	1.409535	1.463202	**2.772136**	1.595578	**3.196487**	**3.973784**
		Neighborhood	1.520535	**1.838573**	**2.622040**	1.455748	**2.473941**	**2.380059**
		T1 Fuzzy	1.512235	1.304159	1.737224	1.269908	**2.591025**	**2.274447**
		IT2 Fuzzy	1.597392	1.304159	1.207744	0.923282	**2.591025**	1.125319
	Max Dependency	Classical	**2.064114**	**2.692228**	**2.977049**	1.617385	**3.196487**	**3.256158**
		Neighborhood	**1.822968**	**1.934980**	**2.165570**	1.675003	**2.473941**	0.763152
		T1 Fuzzy	1.409535	1.237171	**2.004791**	1.110327	**2.591025**	**2.366554**
		IT2 Fuzzy	1.186203	0.928657	**2.189541**	**2.004884**	1.781779	1.661901
	MRMS	Classical	1.512235	1.6277600	1.663281	1.557314	**2.492290**	0.509217
		Neighborhood	0.476905	0.747506	1.390644	0.830018	1.113038	0.006069
		T1 Fuzzy	0.009859	0.250004	0.317924	0.630056	0.404413	0.367636
		IT2 Fuzzy	(73.62/10, 10.28)	(70.70/11, 9.42)	(84.73/21, 5.38)	(80.33/23, 13.20)	(77.32/31, 10.10)	(87.32/21, 5.63)

Table 4.5: Statistical Significance Test of Different Indices on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features using 10-fold CV

Different Classifiers	Different Criteria	Different Indices	Different Data Sets					
			Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	Max Relevance	Class Separability	1.115011	**2.472168**	**2.073534**	**2.048600**	**6.497386**	**4.822097**
		DB Index	0.584072	**2.130392**	**1.813789**	1.700112	**2.054265**	**5.506915**
		Dunn Index	0.834860	**1.950236**	1.716517	**2.178430**	**2.199172**	**5.152440**
		FFE Index	0.946015	**2.130392**	**2.242609**	**2.264202**	**4.859173**	**5.264211**
	Max Dependency	Class Separability	1.601138	**2.663809**	**2.243969**	**2.323531**	**6.793897**	**5.666531**
		DB Index	1.503348	**2.070083**	**2.243969**	**1.936432**	**3.169158**	**5.152440**
		Dunn Index	1.503348	**2.316990**	1.541996	**1.847185**	**2.925197**	**5.873102**
		FFE Index	**1.841419**	**2.237025**	**2.301652**	**2.080508**	**4.656247**	**5.531408**
	MRMS	Class Separability	0.350547	**2.496602**	1.735401	1.311911	**6.319752**	**3.099046**
		DB Index	0.350892	1.778155	**1.981862**	1.123095	**2.393254**	**2.234430**
		Dunn Index	0.350892	1.652727	1.232803	0.927952	**2.644864**	**2.720209**
		FFE Index	0.673319	**2.032179**	**1.915483**	1.778055	**4.548975**	**3.244724**
		IT2 Fuzzy-Rough	(74.35/13, 11.12)	(71.40/28, 14.25)	(82.98/39, 8.22)	(81.44/16, 11.32)	(82.12/44, 5.83)	(90.21/28, 4.32)
SVM	Max Relevance	Class Separability	0.769180	1.431532	**2.243842**	1.771852	**7.071526**	**3.779873**
		DB Index	0.453990	1.334807	**1.913645**	1.187027	**2.273116**	**3.869679**
		Dunn Index	0.453990	1.016960	**1.891999**	1.444019	**2.657028**	**3.424782**
		FFE Index	0.723855	1.431532	**2.597938**	1.665429	**2.771817**	**4.643088**
	Max Dependency	Class Separability	0.943709	1.548117	**2.450750**	1.558841	**6.854029**	**4.802438**
		DB Index	0.769180	1.604992	**1.814608**	1.369699	**3.689533**	**4.690672**
		Dunn Index	0.723855	1.431532	1.604643	1.226493	**3.983078**	**3.908547**
		FFE Index	1.023394	**2.014468**	**2.450750**	1.719061	**3.596160**	**4.261720**
	MRMS	Class Separability	0.276116	**1.894559**	**2.214598**	1.226493	**6.162518**	**4.025186**
		DB Index	0.211177	1.254171	**2.060543**	1.262381	**2.882704**	**1.916750**
		Dunn Index	0.211177	1.654741	1.713686	1.065282	**3.234626**	**2.168377**
		FFE Index	0.287885	1.413922	**2.117784**	1.344644	**3.893412**	**2.218513**
		IT2 Fuzzy-Rough	(73.68/12, 13.69)	(64.40/20, 10.23)	(83.23/12, 6.33)	(73.21/32, 13.00)	(82.45/24, 5.64)	(92.10/28, 3.34)
C4.5	Max Relevance	Class Separability	0.767321	**2.034721**	**3.169138**	**1.890599**	**6.266074**	**3.875326**
		DB Index	0.512109	**1.811593**	**2.917274**	1.647608	**3.222010**	**3.443560**
		Dunn Index	0.530764	1.519722	**2.570995**	1.512274	**4.641924**	**2.947922**
		FFE Index	0.941441	**2.440618**	**3.423544**	**1.847062**	**5.795141**	**3.326535**
	Max Dependency	Class Separability	1.349614	**2.265115**	**2.988887**	**1.984822**	**6.044280**	**3.406261**
		DB Index	1.321490	**2.500327**	**2.337909**	**1.891965**	**2.980918**	**3.010790**
		Dunn Index	1.151407	**2.433686**	**2.171051**	1.729623	**4.549576**	**3.230214**
		FFE Index	1.445551	**2.818033**	**3.180073**	**2.037430**	**4.896537**	**3.818331**
	MRMS	Class Separability	0.312427	**1.937398**	**2.283456**	1.512662	**5.251195**	1.173032
		DB Index	0.209981	1.152782	**2.571924**	1.110327	**3.090949**	0.058717
		Dunn Index	0.234151	1.253903	**2.007133**	1.193101	**4.927627**	0.350821
		FFE Index	0.312427	1.627760	**2.613611**	1.480552	**5.217514**	0.005953
		IT2 Fuzzy-Rough	(73.62/10, 10.28)	(70.70/11, 9.42)	(84.73/21, 5.38)	(80.33/23, 13.20)	(77.32/31, 10.10)	(87.32/21, 5.63)

Table 4.6: Statistical Significance Test of Different Methods on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features using 10-fold CV

Different Classifiers	Different Methods / Algorithms	Different Data Sets					
		Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	InfoGain	0.977065	1.551380	**3.390931**	1.338564	1.491308	**6.443328**
	Classical mRMR	1.047990	1.620809	**3.390931**	1.647540	1.485272	**3.205826**
	Fuzzy-Rough mRMR	0.399542	**1.940375**	0.039378	**2.131915**	0.021473	**1.842660**
	RELIEF	1.458845	**2.280527**	**2.383994**	1.485305	1.308022	**2.820833**
	SIMBA	1.388513	**1.899749**	1.609474	1.233291	1.556904	**2.477535**
	PCA	0.916887	1.234003	0.846798	1.018161	**2.355848**	**2.277635**
	ICA	1.046068	1.168829	0.635750	1.194376	1.401507	1.768760
	LDA	1.291617	1.359409	0.784240	0.837576	1.496231	1.728355
	IT2 Fuzzy-Rough MRMS	(74.35/13, 11.12)	(71.40/28, 14.25)	(82.98/39, 8.22)	(81.44/16, 11.32)	(82.12/44, 5.83)	(90.21/28, 4.32)
SVM	InfoGain	0.777600	0.462668	1.456195	1.342132	1.266113	**3.673329**
	Classical mRMR	0.230758	0.581447	**2.043126**	1.144907	1.574760	**2.891169**
	Fuzzy-Rough mRMR	0.078632	0.291520	1.743176	0.615555	0.166605	1.235803
	RELIEF	1.442813	1.218712	**2.509743**	0.912570	1.039875	**2.388484**
	SIMBA	1.191206	1.178482	**2.345561**	0.915332	1.032586	**2.164768**
	PCA	0.992521	0.450821	**1.878098**	0.569672	0.510479	**2.322164**
	ICA	1.189354	0.628829	**2.259880**	0.435037	-0.334759	**2.093648**
	LDA	1.145922	0.545486	1.668932	0.370508	0.143624	**2.265551**
	IT2 Fuzzy-Rough MRMS	(73.68/12, 13.69)	(64.40/20, 10.23)	(83.23/12, 6.33)	(73.21/32, 13.00)	(82.45/24, 5.64)	(92.10/28, 3.34)
C4.5	InfoGain	1.512235	**2.438845**	**4.105811**	1.512662	1.339808	**2.573666**
	Classical mRMR	0.255446	**2.512334**	**2.187112**	1.251975	1.259542	**1.858234**
	Fuzzy-Rough mRMR	0.086971	**2.254451**	**1.921494**	0.944508	0.968008	**1.861768**
	RELIEF	1.350511	**2.193704**	**3.150230**	1.107989	1.495424	**1.902529**
	SIMBA	1.063305	**2.111040**	**2.692756**	1.110327	1.321386	**1.885663**
	PCA	0.767321	0.925022	**1.913975**	0.830018	**2.535218**	0.531772
	ICA	0.999345	0.961054	**2.362010**	0.811081	-0.895889	-0.140506
	LDA	0.613363	1.533339	**1.959618**	0.935127	-1.320705	-0.647901
	IT2 Fuzzy-Rough MRMS	(73.62/10, 10.28)	(70.70/11, 9.42)	(84.73/21, 5.38)	(80.33/23, 13.20)	(77.32/31, 10.10)	(87.32/21, 5.63)

Table 4.7: Bootstrap $\mathcal{E}_{\text{BS}}^{.632}$ Analysis of Different Rough Sets on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features

Different Classifiers	Different Criteria	Different Rough Sets	Different Data Sets					
			Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	Max Relevance	Classical	65.3/7	69.1/20	67.8/13	72.6/15	64.4/17	76.1/15
		Neighborhood	67.3/9	71.2/19	74.2/15	74.5/25	79.2/10	77.5/13
		T1 Fuzzy	67.2/16	67.4/10	79.7/14	71.4/15	72.4/24	83.4/15
		IT2 Fuzzy	68.1/17	67.8/7	81.9/26	76.2/9	79.3/12	84.3/15
	Max Dependency	Classical	63.9/21	67.3/11	80.3/18	64.5/9	59.1/22	77.1/25
		Neighborhood	72.4/11	62.4/18	81.6/21	72.0/18	70.9/20	77.7/17
		T1 Fuzzy	73.3/25	67.7/16	83.5/12	74.8/9	74.5/12	79.8/19
		IT2 Fuzzy	73.3/25	71.2/12	83.5/12	74.8/8	75.2/22	79.9/25
	MRMS	Classical	58.2/24	71.2/25	84.2/13	77.1/17	77.6/24	81.5/9
		Neighborhood	70.4/27	72.2/19	87.3/14	**90.3/8**	81.2/21	84.8/16
		T1 Fuzzy	81.1/24	72.8/24	87.0/22	79.3/10	80.3/20	82.0/19
		IT2 Fuzzy	**85.1/43**	**75.7/42**	**90.7/37**	88.8/39	**82.9/48**	**84.9/48**
SVM	Max Relevance	Classical	68.2/23	69.3/12	74.5/32	65.2/20	70.0/30	85.4/38
		Neighborhood	68.6/13	71.0/14	75.4/38	65.7/19	75.4/26	83.0/38
		T1 Fuzzy	69.4/16	72.9/11	77.6/32	69.6/16	78.3/25	84.7/38
		IT2 Fuzzy	70.0/16	72.9/20	77.6/44	69.9/29	79.9/33	86.6/25
	Max Dependency	Classical	69.5/21	69.5/10	71.8/27	64.4/15	70.5/23	85.5/11
		Neighborhood	70.1/8	71.4/11	73.3/31	67.6/14	74.8/6	82.0/14
		T1 Fuzzy	70.8/15	72.0/15	74.8/14	68.5/9	78.5/17	84.1/21
		IT2 Fuzzy	71.2/22	75.4/11	75.5/24	69.0/19	81.5/31	86.9/28
	MRMS	Classical	69.8/20	75.0/12	76.4/37	63.8/14	74.2/47	87.6/44
		Neighborhood	70.0/23	74.7/17	76.8/40	64.8/19	79.3/42	85.8/39
		T1 Fuzzy	71.2/17	75.7/10	80.0/42	66.3/9	82.1/46	87.3/42
		IT2 Fuzzy	**71.2/23**	**76.6/31**	**80.8/42**	**71.1/28**	**82.1/46**	**88.9/46**
C4.5	Max Relevance	Classical	76.0/17	78.3/16	74.1/34	67.1/15	62.3/42	81.6/27
		Neighborhood	76.0/12	78.9/14	74.6/25	69.7/10	67.5/30	81.3/23
		T1 Fuzzy-Rough	78.5/10	79.0/21	77.2/30	74.0/13	70.7/41	78.7/34
		IT2 Fuzzy-Rough	79.6/19	80.5/9	77.1/27	73.2/17	73.5/32	81.2/38
	Max Dependency	Classical	77.1/10	79.3/9	71.1/25	66.8/15	62.0/27	81.1/14
		Neighborhood	77.6/23	80.0/18	73.1/15	69.5/10	66.2/18	81.8/26
		T1 Fuzzy-Rough	78.4/12	80.1/16	74.2/8	72.9/14	70.5/11	81.3/18
		IT2 Fuzzy-Rough	80.8/6	81.7/16	74.9/20	73.2/19	72.2/26	82.5/15
	MRMS	Classical	78.1/21	82.0/18	75.7/37	64.4/20	66.2/37	80.6/44
		Neighborhood	78.1/20	82.6/16	76.0/38	67.2/14	72.0/39	80.7/40
		T1 Fuzzy-Rough	80.8/17	82.4/11	79.7/41	69.7/20	75.0/44	81.7/42
		IT2 Fuzzy-Rough	**81.1/34**	**82.6/22**	**80.8/45**	**74.6/30**	**77.3/33**	**83.5/46**

Table 4.8: Bootstrap $\mathcal{E}_{\text{BS}}^{.632}$ Analysis of Different Indices on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features

Different Classifiers	Different Criteria	Different Indices	Different Data Sets					
			Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	Max Relevance	Class Separability	70.3/24	63.8/11	80.4/9	68.7/22	67.8/21	74.3/25
		DB Index	73.9/21	69.8/14	82.4/20	66.8/15	71.3/24	80.9/26
		Dunn Index	71.6/9	63.9/17	83.3/24	66.2/15	77.4/19	77.9/14
		FFE Index	73.4/19	68.7/11	77.2/19	65.9/27	65.1/10	82.5/16
	Max Dependency	Class Separability	69.9/14	72.3/20	85.2/23	70.7/11	73.9/14	75.7/18
		DB Index	67.2/8	68.5/8	87.1/16	80.6/10	74.9/20	78.0/13
		Dunn Index	68.0/14	67.2/22	82.4/8	87.2/18	76.0/18	79.6/9
		FFE Index	64.4/21	66.1/22	86.6/11	84.7/20	75.9/14	78.7/17
	MRMS	Class Separability	79.9/7	69.6/12	81.1/12	82.1/13	74.4/26	80.6/19
		DB Index	80.0/19	74.9/22	87.8/21	**89.6/12**	78.3/25	77.0/8
		Dunn Index	80.9/9	75.3/8	88.7/18	86.3/26	79.2/8	84.0/14
		FFE Index	84.9/13	73.4/15	85.2/19	84.9/18	80.3/22	84.6/9
		IT2 Fuzzy-Rough MRMS	**85.1/43**	**75.7/42**	**90.7/37**	88.8/39	**82.9/48**	**84.9/48**
SVM	Max Relevance	Class Separability	68.0/17	68.3/10	70.3/32	62.3/13	51.2/28	73.9/25
		DB Index	66.6/16	68.3/16	70.5/45	65.0/16	70.3/39	79.0/41
		Dunn Index	65.7/22	69.0/10	71.9/33	62.1/10	68.1/39	79.2/32
		FFE Index	67.2/16	67.6/20	69.9/43	61.9/32	66.6/44	76.0/37
	Max Dependency	Class Separability	67.3/18	68.1/19	68.6/15	61.4/16	53.3/12	73.4/27
		DB Index	66.6/7	69.2/11	67.8/14	61.8/12	71.2/24	78.5/25
		Dunn Index	67.2/15	68.6/21	70.1/14	62.6/12	67.6/21	78.2/18
		FFE Index	68.8/22	68.3/19	67.8/27	61.6/15	67.7/15	75.0/19
	MRMS	Class Separability	69.4/12	74.2/11	73.1/40	62.4/12	65.0/38	76.3/43
		DB Index	68.1/11	74.8/23	73.9/42	61.9/15	72.7/40	81.9/48
		Dunn Index	67.3/9	73.0/15	75.8/30	61.6/10	71.8/45	81.9/44
		FFE Index	68.5/13	63.9/18	74.1/46	61.4/13	69.1/43	78.7/50
		IT2 Fuzzy-Rough MRMS	**71.2/23**	**76.6/31**	**80.8/42**	**71.1/28**	**82.1/46**	**88.9/46**
C4.5	Max Relevance	Class Separability	73.9/16	78.0/19	70.1/43	67.4/14	50.6/33	76.0/38
		DB Index	75.0/9	77.3/14	70.3/29	68.8/18	60.0/28	80.5/40
		Dunn Index	73.9/22	74.9/17	71.7/36	69.8/10	50.9/37	80.7/30
		FFE Index	68.2/10	72.9/18	69.5/36	67.8/17	51.7/31	80.2/32
	Max Dependency	Class Separability	73.9/20	77.4/14	67.9/27	66.4/12	50.2/27	75.2/29
		DB Index	76.5/22	78.2/21	67.2/34	67.9/17	58.7/29	79.7/32
		Dunn Index	75.1/19	74.0/16	69.6/32	67.0/16	49.4/34	80.1/32
		FFE Index	69.0/20	71.8/22	67.4/29	66.8/20	51.1/28	80.2/27
	MRMS	Class Separability	76.4/13	81.8/19	72.9/39	57.6/18	55.2/40	78.9/43
		DB Index	77.1/15	81.2/12	73.3/36	61.3/19	63.5/39	80.3/39
		Dunn Index	76.0/21	78.5/17	75.5/46	66.2/16	54.7/14	80.2/44
		FFE Index	70.3/15	71.9/16	73.2/42	63.2/15	55.3/40	81.6/48
		IT2 Fuzzy-Rough MRMS	**81.1/34**	**82.6/22**	**80.8/45**	**74.6/30**	**77.3/33**	**83.5/46**

Table 4.9: Bootstrap $\mathcal{E}_{\text{BS}}^{.632}$ Analysis of Different Methods on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features

Different Classifiers	Different Methods / Algorithms	Different Data Sets					
		Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	InfoGain	75.1/27	67.1/8	70.2/25	83.8/25	78.6/25	76.5/11
	Classical mRMR	75.4/24	73.4/15	75.1/21	73.1/26	72.8/25	75.5/22
	Fuzzy-Rough mRMR	83.9/15	74.9/18	81.7/20	85.7/20	81.7/14	78.5/18
	RELIEF	75.6/20	65.6/9	81.7/9	86.7/25	80.3/17	82.0/11
	SIMBA	71.7/26	73.9/16	74.9/15	79.9/14	80.2/12	82.8/12
	PCA	83.1/23	73.3/14	85.6/7	85.7/18	81.9/27	84.2/19
	ICA	83.5/9	73.7/14	79.5/26	81.9/18	80.2/9	81.5/16
	LDA	84.0/23	72.6/18	86.2/24	79.3/25	81.2/23	82.7/15
	IT2 Fuzzy-Rough MRMS	**85.1/43**	**75.7/42**	**90.7/37**	**88.8/39**	**82.9/48**	**84.9/48**
SVM	InfoGain	65.6/23	67.3/12	63.4/41	57.6/11	76.5/35	77.8/39
	Classical mRMR	70.4/19	66.7/18	73.7/39	61.3/23	75.6/36	83.0/46
	Fuzzy-Rough mRMR	71.0/11	71.4/24	75.3/29	66.2/24	81.9/36	87.7/35
	RELIEF	64.4/22	66.5/18	71.9/32	63.2/15	79.2/38	83.8/40
	SIMBA	65.6/12	67.2/19	71.7/29	64.4/21	79.4/39	84.9/29
	PCA	65.8/12	70.6/12	74.5/45	67.2/14	80.8/36	83.3/40
	ICA	64.3/9	68.4/10	73.4/39	66.7/12	**83.3/38**	83.1/36
	LDA	65.5/10	68.1/8	74.7/36	67.5/10	82.1/43	85.3/34
	IT2 Fuzzy-Rough MRMS	**71.2/23**	**76.6/31**	**80.8/42**	**71.1/28**	82.1/46	**88.9/46**
C4.5	InfoGain	70.7/11	73.2/16	69.8/36	67.5/23	69.9/40	77.8/37
	Classical mRMR	78.2/12	72.5/16	72.8/32	67.5/16	70.7/37	79.2/35
	Fuzzy-Rough mRMR	78.8/21	77.2/11	74.9/38	70.7/17	72.4/35	80.8/27
	RELIEF	69.8/17	71.3/14	71.6/36	68.1/17	70.5/27	79.9/49
	SIMBA	71.5/10	72.3/10	71.7/35	68.3/19	70.8/38	79.0/44
	PCA	72.9/10	75.7/11	73.5/37	69.5/14	65.4/39	80.2/39
	ICA	71.3/10	72.5/12	73.3/45	71.3/12	80.1/41	80.8/41
	LDA	74.6/11	73.8/13	74.0/34	71.3/14	**81.8/36**	81.1/38
	IT2 Fuzzy-Rough MRMS	**81.1/34**	**82.6/22**	**80.8/45**	**74.6/30**	77.3/33	**83.5/46**

4.4.2 Performance of Various Rough Set Based Models

In case of dimensionality reduction, the reduced feature set is always relative to a certain feature evaluation index. In general, different evaluation indices may lead to different reduced feature subsets. To establish the effectiveness of IT2 fuzzy-rough sets over Pawlak's or classical rough sets, neighborhood rough sets, and T1 fuzzy-rough sets, extensive experiments are done on various data sets. Different feature evaluation criteria such as Max-Dependency, Max-Relevance, and MRMS are considered for feature selection. In this regard, it should be mentioned that the classical rough sets and FR-MRMS set based feature selection methods reported in [327] and [315], respectively, use the MRMS criterion, while quick reduct [83], neighborhood quick reduct [267], fuzzy-rough quick reduct [220], and IT2 fuzzy-rough quick reduct [531] algorithms select features using Max-Dependency criterion.

Table 3.1 of Chapter 3 and Tables 4.3, 4.4, and 4.7 present the comparative performance of different rough set models for attribute selection task. The results and subsequent discussions are presented in these tables with respect to the classification accuracy of the K-NN, SVM, and C4.5. The variations of bootstrap error $\mathcal{E}_{BS}^{632}$ for different rough sets using MRMS criteria with respect to number of features selected are also represented in Figure 7.4. From the results reported in Table 3.1 of Chapter 3 and Table 4.3, it can be seen that the IT2FR-MRMS method attains maximum classification accuracy of the K-NN, SVM, and C4.5 in most of the cases. Out of 12 cases of training-testing, the IT2FR-MRMS method achieves highest classification accuracy in 10 cases, while FR-MRMS method attains it only in 2 cases. Similarly, from the results reported in Table 4.7, it can be seen that the IT2FR-MRMS method attains maximum classification accuracy of the K-NN, SVM and C4.5 in all cases, except one. On the other hand, among the 198 comparisons of 10-fold CV reported in Table 4.4, which are calculated using Behrens-Fisher method of hypothesis testing (equation (3.15) of Chapter 3), the IT2FR-MRMS provides significantly better results in 64 cases and better results but not significantly in all other cases. In brief, out of total 48 cases, the T2 fuzzy-rough MRMS method attains highest classification accuracy in 45 cases, irrespective of the data sets and classifiers used.

Following conclusions can also be drawn from the results reported in Table 3.1 of Chapter 3, Tables 4.3, 4.4, 4.7 and Figure 7.4: (i) the performance of the MRMS criterion is significantly better than that of other criteria, namely, Max-Dependency and Max-Relevance, irrespective of the rough set models used; (ii) the performance of the IT2 fuzzy-rough sets is significantly better than that of the T1 fuzzy-rough sets as well as classical and neighborhood rough sets in almost all the cases, irrespective of the feature evaluation criteria used; and (iii) the IT2FR-MRMS method using the IT2 fuzzy-rough sets achieves highest classification accuracy in most of the cases, irrespective of the data

sets, feature evaluation criteria, rough set models, experimental setup, and classifiers used.

The better performance of the IT2FR-MRMS method is achieved due to the fact that the MRMS criterion can identify relevant and significant features from high dimensional real life data sets more efficiently than Max-Dependency and Max-Relevance criteria, while the IT2 fuzzy-rough sets can capture uncertainties associated with data more accurately.

4.4.3 Performance of Various Feature Evaluation Indices

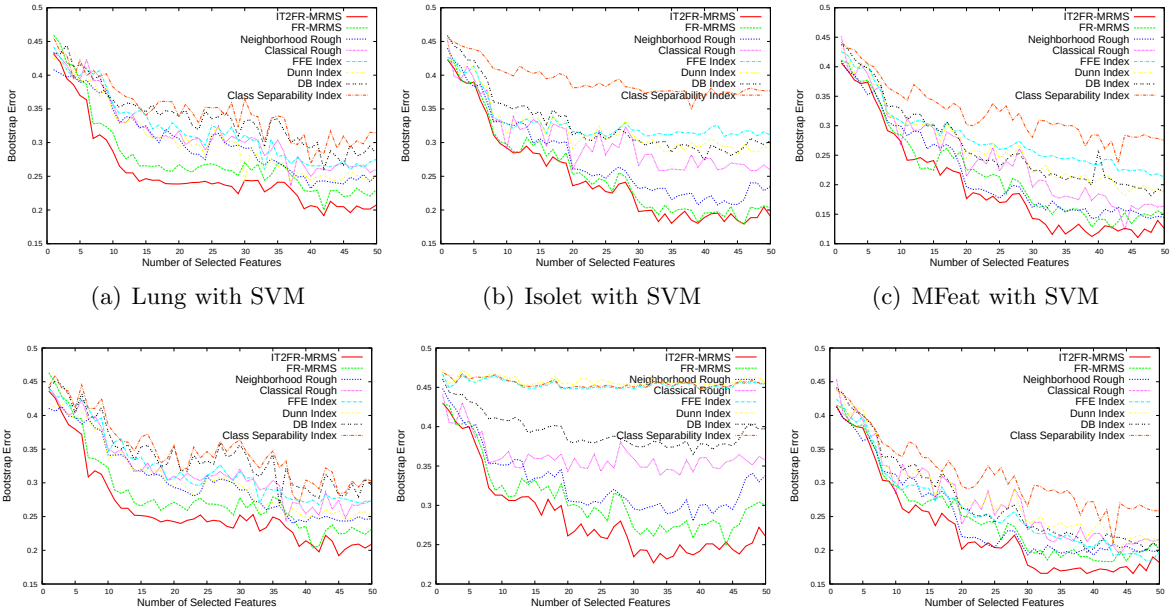
In order to establish the effectiveness of the IT2 fuzzy-rough sets over other feature evaluation indices, extensive experimentation is done on different real life data sets. Table 3.1 of Chapter 3, Tables 4.3, 4.5, and 4.8 present the comparative performance of the IT2FR-MRMS method and various feature evaluation indices such as class separability index, DB index, Dunn index and FFE index considering different feature evaluation criteria such as Max-Relevance, Max-Dependency, and MRMS. The variations of bootstrap error $\mathcal{E}_{BS}^{.632}$ for IT2FR-MRMS and different indices using MRMS criteria with respect to number of features selected are represented in Figure 7.4.

From the results reported in Table 3.1 of Chapter 3, Tables 4.3, and 4.8, it can be seen that the IT2 fuzzy-rough MRMS method attains highest classification accuracy in all the cases, except one, on Satimage, Segmentation, Leukemia II, Breast II Colon, Breast I, Leukemia I, Isolet, and Multiple Features data sets, irrespective of the classifiers used. Table 4.5 reports the comparative performance in case of 10-fold CV. All the results reported in Table 4.5 confirms that the IT2FR-MRMS method provides significantly better results in 128 cases and better but not significantly in all other cases out of total 216 comparisons. The results reported in Table 3.1 of Chapter 3, Tables 4.3, 4.5, 4.8 and Figure 7.4 confirm the fact that the performance of the MRMS criterion is better than that of Max-Dependency and Max-Relevance, irrespective of feature evaluation indices.

4.4.4 Performance of Different Algorithms

Finally, Table 3.1 of Chapter 3, Tables 4.3, 4.6, and 4.9 compare the performance of the proposed IT2FR-MRMS algorithm with that of different existing feature selection and extraction algorithms on various data sets. From the results reported in Table 3.1 of Chapter 3 and Table 4.3, it is seen that the IT2FR-MRMS algorithm achieves highest classification accuracy of the K-NN, SVM, and C4.5 in 10 cases out of total 12 cases, while the PCA and fuzzy-rough mRMR attain highest classification accuracy in only 1 and 1 cases, respectively.

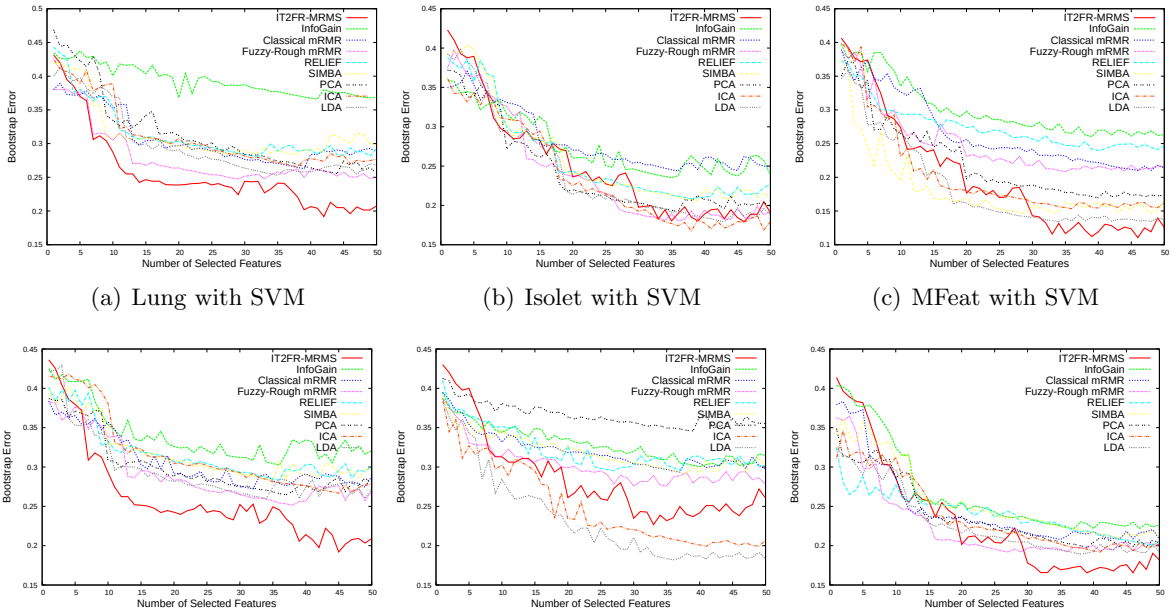
Table 4.6 reports the performance of different methods in case of 10-fold CV, along with the results of test of significance, for the K-NN, SVM and C4.5. From the results reported in these tables, it can be seen that the proposed method attains significantly



(a) Lung with SVM (b) Isolet with SVM (c) MFeat with SVM

(d) Lung with C4.5 (e) Isolet with C4.5 (f) MFeat with C4.5

Figure 4.9: Variation of bootstrap error $\mathcal{E}_{BS}^{.632}$ for different rough sets and indices using MRMS criteria with respect to number of features selected.



(a) Lung with SVM (b) Isolet with SVM (c) MFeat with SVM

(d) Lung with C4.5 (e) Isolet with C4.5 (f) MFeat with C4.5

Figure 4.10: Variation of bootstrap error $\mathcal{E}_{BS}^{.632}$ for IT2FR-MRMS and other methods with respect to number of features selected.

better results than other algorithms in 45 cases out of total 144 cases and better results but not significantly in 94 cases. On the other hand, ICA and LDA achieve better results in 3 and 2 cases, respectively, but not significantly. In brief, out of total 48 cases, the IT2FR-MRMS method attains highest classification accuracy in 41 cases, while the PCA, ICA, LDA, and fuzzy-rough mRMR methods achieve it only 1, 2, 3, and 1 cases, respectively. Also, Table 4.9 confirms that the IT2FR-MRMS algorithm achieves highest classification accuracy of both SVM and C4.5 in 16 cases out of total 18 cases, while the ICA and LDA attain highest classification accuracy in only 1 and 1 cases, respectively. The variations of bootstrap error $\mathcal{E}_{BS}^{.632}$ for IT2FR-MRMS and other methods with respect to number of features selected are represented in Figure 4.10.

All the results reported in Tables 3.1 of Chapter 3, Table 4.3, 4.6, 4.9 and Figure 4.10 confirm that the proposed IT2FR-MRMS method selects a feature set having highest classification accuracy of K-NN, SVM and C4.5 in most of the cases, irrespective of the data sets. The proposed method can potentially yield significantly better results than the existing algorithms. The better performance of the proposed method is achieved due to the fact that it provides an efficient way to select a reduced feature set having maximum relevance and significance.

4.4.5 Execution Time

A reduced set of relevant and significant features is obtained using the IT2FR-MRMS algorithm with significantly lesser time. For Max-Relevance, IT2 fuzzy-rough sets takes 2.12E+01, 1.83E+01, 2.53E+01, 5.23E+01, 0.22, 2.83, 9.32, 2.45, 2.67E+01, and 7.82 seconds, and for Max-Dependency, it takes 3.62E+04, 2.93E+04, 5.88E+05, 1.24E+06, 9.98E+03, 1.01E+04, 6.12E+05, 9.88E+03, 2.93E+05, 5.43E+04 seconds, respectively, for Satimage, Segmentation, Leukemia II, Breast II, Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features data sets, whereas the execution time for the other algorithms compared are reported in Table 3.8. The proposed IT2FR-MRMS took only 4.58E+01, 2.55E+01, 2.82E+01, 6.43E+01, 0.58, 2.92, 9.88, 3.13, 3.63E+01, and 1.33E+01 seconds in case of Satimage, Segmentation, Leukemia II, Breast II, Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features respectively, which are lesser or comparable with the execution times of other algorithms that produce relatively good results.

4.5 Conclusion

The major contribution of this chapter is three fold, namely,

1. development of a new feature selection method, integrating judiciously the theory of IT2 fuzzy-rough sets and merit of the MRMS criterion;

2. introduction of the concepts of LFEPM and UFEPM to compute both relevance and significance in IT2 fuzzy approximation spaces; and

3. application of the proposed method in selecting discriminative and significant features from high dimensional benchmark and microarray gene expression data sets.

The performance of the proposed feature selection method is compared with that of other feature selection and extraction methods using the predictive accuracy of nearest neighbor rule, support vector machine, and decision tree. The proposed method uses the concept of IT2 fuzzy-rough feature relevance and significance for finding significant and relevant features of real valued data sets. This formulation is geared towards maximizing the utility of IT2 fuzzy-rough sets, feature selection, and the MRMS criterion with respect to knowledge discovery tasks. Through these investigations and experiments, the potential utility of IT2 fuzzy-rough sets and the MRMS criterion for attribute selection is demonstrated.

The results obtained on different benchmark and microarray data sets demonstrate that a feature extraction technique such as PCA, ICA, or LDA may provide a reacher feature subset than that obtained using a feature selection algorithm with a higher cost. However, it is very difficult to decide whether to select a feature from original measurement space or extract a new feature by transforming the existing features for a given data set. Hence, a dimensionality reduction algorithm needs to be formulated that can simultaneously select or extract features depending upon the criteria, integrating the merits of both feature selection and extraction techniques. In this regard, a novel dimensionality reduction method, based on fuzzy-rough sets, is presented in the next chapter, that will simultaneously select and extract features using the concept of feature significance.

Chapter 5

Simultaneous Attribute Selection and Feature Extraction

5.1 Introduction

In Chapter 3 and Chapter 4, some basic concepts of dimensionality reduction have already been discussed, which is a process of selecting a map by which a sample in an m-dimensional measurement space is transformed into an object in a d-dimensional feature space, where $d < m$ [101]. The problem of dimensionality reduction has two aspects, namely, formulation of a suitable criterion to evaluate the goodness of a feature set and searching the optimal set in terms of the criterion [169]. In general, those features are considered to have optimal saliencies for which interclass (respectively, intraclass) distances are maximized (respectively, minimized). The criterion of a good feature is that, it should be unchanging with any other possible variation within a class, while emphasizing on differences that are important in discriminating between patterns of different classes [525].

The major mathematical measures devised so far for the estimation of feature quality are mostly statistical in nature, and can be broadly classified into two categories, namely, feature selection in the measurement space and feature selection in a transformed space. The techniques in the first category generally reduce the dimensionality of the measurement space by discarding redundant or least information carrying features [61]. On the other hand, those in the second category utilize all the information contained in the measurement space to obtain a new transformed space, thereby mapping a higher dimensional pattern to a lower dimensional one. This is referred to as feature extraction [101, 112].

An optimal feature subset, selected or extracted by a dimensionality reduction method, is always relative to a certain feature evaluation criterion. In general, different criteria may lead to different optimal feature subsets. However, every criterion tries to measure the discriminating ability of a feature or a subset of features to distinguish different class

labels. One of the main problems in real life data analysis is uncertainty. Some of the sources of this uncertainty include incompleteness and vagueness in class definitions. In this background, the possibility concept introduced by rough set theory [414] has gained popularity in modeling and propagating uncertainty. It has been applied to reasoning with uncertainty, fuzzy rule extraction and modeling, classification, clustering, and feature selection [325, 414].

Rough sets can be used to find most informative feature subset of original attributes from a given data with discretized attribute values [83, 327, 400]. However, there are usually real valued data and fuzzy information in real world applications. In rough sets, the real valued features are divided into several discrete partitions and the dependency or quality of approximation of a feature is calculated. The inherent error that exists in discretization process is of major concern in the computation of the dependency of real valued features. Combining fuzzy and rough sets provides an important direction in reasoning with uncertainty for real valued data [110, 220, 571]. They are complementary in some aspects. The generalized theories of rough-fuzzy computing have been applied successfully to feature selection of real valued data set [76, 199, 220, 223, 323, 325, 500]. Also, neighborhood rough sets [197, 198] are found to be suitable for both numerical and categorical data set. In [198], Hu et al. described a neighborhood rough set based feature selection algorithm.

On the other hand, a feature extraction technique such as principal component analysis, linear discriminant analysis, and independent component analysis [112], generates a new set of features using a mapping function that takes some linear or nonlinear combination of original features. While principal component analysis uses a linear orthogonal transformation to project a sample space containing possibly correlated variables into a different space with uncorrelated variables, independent component analysis decomposes a multidimensional feature vector into statistically independent components to reveal the hidden factors from a set of random variables [112].

In general, a feature extraction technique provides a reacher feature subset than that obtained using a feature selection algorithm with a higher cost [315]. Hence, it is very difficult to decide whether to select a feature from original measurement space or extract a new feature by transforming the existing features for a given data set. A dimensionality reduction algorithm needs to be formulated that can simultaneously select or extract features depending upon the criteria, integrating the merits of both feature selection and extraction techniques.

In this regard, a novel dimensionality reduction algorithm is proposed based on fuzzy-rough sets (FR-SFSFEx), which simultaneously selects and extracts features from a given data set. Using the concept of feature significance, the feature set in each iteration is partitioned into three subsets, namely, insignificant, dispensable, and significant feature

sets. The insignificant feature set is discarded from the current feature set, while significant feature set is used to select or extract a feature in next iteration. Depending on the quality of features present in the dispensable set of current iteration, a new feature is extracted or an existing feature is selected from the dispensable set for reduced feature set. In effect, the finally obtained reduced feature set may simultaneously contain some original features of measurement space and extracted new features of transformed space, which are both relevant and significant. The effectiveness of the proposed fuzzy-rough dimensionality reduction method, along with a comparison with other methods, is demonstrated on a set of real life data using the predictive accuracy of nearest neighbor rule, support vector machine, and decision tree. Some of the results, presented in this chapter, are also reported in [314, 317] .

The structure of the rest of this chapter is as follows: Section 5.2 describes the proposed fuzzy-rough set based simultaneous feature selection and extraction method. A few case studies and a comparison with other methods are presented in Section 5.3. Concluding remarks are given in Section 5.4.

5.2 Fuzzy-Rough Sets for Dimensionality Reduction

In this section, a new dimensionality reduction method is presented, termed as FR-SFSFEx, integrating the theory of fuzzy-rough sets and merits of both feature selection and extraction techniques. The method is based on the concept of feature significance, which has been discussed in Section 3.2.2 of Chapter 3.

5.2.1 Simultaneous Feature Selection and Extraction

The high dimensional real life data set may generally contain a number of nonrelevant and insignificant features. The presence of such features may lead to a reduction in the useful information. Ideally, the reduced feature set obtained using a dimensionality reduction algorithm should contain features those have high relevance with the classes while the significance among them would be as high as possible. The relevant and significant features are expected to be able to predict the respective classes of the samples. Hence, to assess the effectiveness of the features, both relevance and significance, described in Section 3.2.2 of Chapter 3, need to be measured quantitatively. The proposed dimensionality reduction method addresses the above issues through following three phases:

1. computation of the relevance of each feature present in original feature set;
2. determination of the insignificant, dispensable, and significant feature sets; and
3. extraction of a relevant feature from the dispensable set.

The fuzzy-rough set is used to compute both relevance and significance of features. The insignificant feature set is discarded from the whole feature set, while the significant feature set is used to select or extract significant features for reduced feature set.

Let $\gamma_{\mathcal{A}_i}(\mathbb{D})$ represents the relevance of feature $\mathcal{A}_i \in \mathbb{C}$. The proposed algorithm starts with a single feature $\mathcal{A}_i$ that has the highest relevance value. Based on the significance values of all other features, the feature set $\mathbb{C}$ is then partitioned into three subsets, namely, insignificant set I_i, dispensable set D_i, and significant set S_i, which are defined as follows:

$$\mathrm{I}_i = \{\mathcal{A}_j | \sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathcal{A}_j, \mathbb{D}) < -\delta_i; \mathcal{A}_j \neq \mathcal{A}_i \in \mathbb{C}\}; \tag{5.1}$$

$$\mathrm{D}_i = \{\mathcal{A}_j | -\delta_i \leq \sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathcal{A}_j, \mathbb{D}) \leq \delta_i; \mathcal{A}_j \neq \mathcal{A}_i \in \mathbb{C}\}; \tag{5.2}$$

$$\mathrm{S}_i = \{\mathcal{A}_j | \sigma_{\{\mathcal{A}_i,\mathcal{A}_j\}}(\mathcal{A}_j, \mathbb{D}) > \delta_i; \mathcal{A}_j \neq \mathcal{A}_i \in \mathbb{C}\}; \tag{5.3}$$

where $\mathbb{C} = \mathrm{I}_i \cup \mathrm{D}_i \cup \mathrm{S}_i \cup \{\mathcal{A}_i\}$ and δ_i is a pre-defined threshold value corresponding to the feature $\mathcal{A}_i$.

The insignificant set I_i represents the set of features those are insignificant with respect to the candidate feature $\mathcal{A}_i$ of the current iteration. Hence, the insignificant set I_i should be discarded from the whole feature set $\mathbb{C}$ as the presence of such insignificant features may lead to a reduction in the useful information. If insignificant features are present in the reduced feature set, they may reduce the classification or clustering performance. The significant set S_i consists of set of features those are significant with respect to the feature $\mathcal{A}_i$. In other words, the set S_i represents the set of features of $\mathbb{C}$ those have the significance values with respect to the feature $\mathcal{A}_i$ greater than the threshold δ_i. This set is considered in the next iteration to select or extract a new feature.

On the other hand, the dispensable set D_i is used for extracting a new feature in the current iteration. As the significance values of the features present in the dispensable set are very low, they form a group of similar features. These features may be considered to generate a new feature. However, the similar features of dispensable set may be in phase or out of phase with respect to each other. Hence, the following definition can be used to extract a new feature $\overline{\mathcal{A}}_i$ from the dispensable set of features D_i:

$$\overline{\mathcal{A}}_i = \frac{\mathcal{A}_i + \sum \lambda_j \mathcal{A}_j}{1 + \sum |\lambda_j|}; \tag{5.4}$$

where $\lambda_j \in \{-1, 0, 1\}$ and $\mathcal{A}_j \in \mathrm{D}_i$.

To find out the value of λ_j for each feature $\mathcal{A}_j \in \mathrm{D}_i$, the following greedy algorithm can be used. Let $\mathcal{A}_i$ be the initial representative of the set D_i. The representative of D_i is refined incrementally. By searching among the features of set D_i, the current representative is merged and averaged with other features, both in phase and out of phase, such that the augmented representative $\overline{\mathcal{A}}_i$ increases the relevance value. The merging process is

repeated until the relevance value can no longer be improved. If a feature $\mathcal{A}_j \in \mathrm{D}_i$ in phase (respectively, out of phase) with the feature $\mathcal{A}_i$ increases the relevance value, then $\lambda_j = 1$ (respectively, $\lambda_j = -1$). On the other hand, the value of $\lambda_j = 0$ if feature $\mathcal{A}_j$ does not increase the relevance value, irrespective of the phases. The main steps to find out the values of λ_j for all $\mathcal{A}_j \in \mathrm{D}_i$ are as follows:

1. Initialize $\bar{\mathrm{D}}_i \leftarrow \{\mathcal{A}_i\}$, $\gamma_{\max} \leftarrow \gamma_{\mathcal{A}_i}$, and $\lambda_i = 1$.
2. Repeat following six steps (steps 3 to 8) until $\mathrm{D}_i = \emptyset$.
3. Initialize max $\leftarrow 0$ and $\lambda_j = 0, \forall \mathcal{A}_j \in \mathrm{D}_i$.
4. Repeat the following three steps (steps 5 to 7) for each feature $\mathcal{A}_j \in \mathrm{D}_i$.
5. Compute two augmented representatives $\mathcal{A}^+_{i+j}$ and $\mathcal{A}^-_{i+j}$ by averaging the features of $\bar{\mathrm{D}}_i$ with $\mathcal{A}_j$ and its complement, respectively, as follows:

$$\mathcal{A}^+_{i+j} = \frac{1}{1 + \sum |\lambda_k|} \left\{ \sum_{\mathcal{A}_k \in \bar{\mathrm{D}}_i} \lambda_k \mathcal{A}_k + \mathcal{A}_j \right\} \tag{5.5}$$

$$\mathcal{A}^-_{i+j} = \frac{1}{1 + \sum |\lambda_k|} \left\{ \sum_{\mathcal{A}_k \in \bar{\mathrm{D}}_i} \lambda_k \mathcal{A}_k - \mathcal{A}_j \right\} \tag{5.6}$$

6. Evaluate the value of λ_j as follows:

$$\lambda_j = \begin{cases} 1 & \text{if } \gamma_{\mathcal{A}^+_{i+j}} \geq \gamma_{\mathcal{A}^-_{i+j}} \text{and } \gamma_{\mathcal{A}^+_{i+j}} > \gamma_{\max} \\ -1 & \text{if } \gamma_{\mathcal{A}^-_{i+j}} \geq \gamma_{\mathcal{A}^+_{i+j}} \text{and } \gamma_{\mathcal{A}^-_{i+j}} > \gamma_{\max} \end{cases}$$

7. If $\lambda_j \neq 0$, then max $\leftarrow j$ and

$$\gamma_{\max} = \begin{cases} \gamma_{\mathcal{A}^+_{i+j}} & \text{if } \lambda_j = 1 \\ \gamma_{\mathcal{A}^-_{i+j}} & \text{if } \lambda_j = -1 \end{cases} \tag{5.7}$$

8. If max $\neq 0$, then $\mathrm{D}_i \leftarrow \mathrm{D}_i \setminus \{\mathcal{A}_{\max}\}$, $\bar{\mathrm{D}}_i \leftarrow \bar{\mathrm{D}}_i \cup \{\mathcal{A}_{\max}\}$; otherwise stop.

After extracting the feature $\overline{\mathcal{A}}_i$ from the dispensable set D_i using (5.4), the insignificant feature set I_i and used features of D_i are discarded from the whole feature set $\mathbb{C}$. From the remaining features of $\mathbb{C}$, another feature $\mathcal{A}_j$ is selected by maximizing the following condition:

$$\gamma_{\mathcal{A}_j}(\mathbb{D}) + \frac{1}{|\mathbb{S}|} \sum_{\overline{\mathcal{A}}_i \in \mathbb{S}} \sigma_{\{\overline{\mathcal{A}}_i, \mathcal{A}_j\}}(\mathcal{A}_j, \mathbb{D}); \tag{5.8}$$

where $\mathbb{S}$ is the already selected or extracted feature set. The process is repeated to select or extract more features. The main steps of the proposed simultaneous feature selection and extraction algorithm are reported next.

5.2.2 FR-SFSFEx Algorithm

- *Input:* Original set $\mathbb{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_k, \ldots, \mathcal{A}_m\}$.
- *Output:* Reduced set $\mathbb{S} = \{\overline{\mathcal{A}}_1, \ldots, \overline{\mathcal{A}}_i, \ldots, \overline{\mathcal{A}}_d\}$.

1. Initialize $\mathcal{B} \leftarrow \{\mathcal{A}_1, \ldots, \mathcal{A}_k, \ldots, \mathcal{A}_m\}$ and $\mathbb{S} \leftarrow \emptyset$.
2. Calculate relevance value $\gamma_{\mathcal{A}_i}(\mathbb{D})$ of feature $\mathcal{A}_i \in \mathcal{B}$.
3. Select feature $\mathcal{A}_i$ from $\mathcal{B}$ as the first feature that has highest relevance value.
4. Repeat the following five steps (steps a to e) until $\mathcal{B} = \emptyset$ or desired number of features are selected.
   (a) Generate three subsets, namely, insignificant set I_i, dispensable set D_i, and significant set S_i with respect to the candidate feature $\mathcal{A}_i$.
   (b) Evaluate the values of λ_j for all $\mathcal{A}_j \in \mathrm{D}_i$.
   (c) Extract feature $\overline{\mathcal{A}}_i$ from the dispensable set D_i using (5.4) and add it to the reduced set $\mathbb{S}$.
   (d) Discard the subset I_i and used features of D_i from the candidate feature set $\mathcal{B}$.
   (e) From the remaining features of $\mathcal{B}$, select feature $\mathcal{A}_j$ that maximizes condition (5.8).
5. Stop.

5.2.3 Fundamental Properties

From the above discussions, the following properties can be stated about the proposed algorithm:

1. The relevance of extracted feature $\overline{\mathcal{A}}_i$ is higher than that of original feature $\mathcal{A}_i$, that is, $\gamma_{\overline{\mathcal{A}}_i}(\mathbb{D}) \geq \gamma_{\mathcal{A}_i}(\mathbb{D})$.
2. The significance $\sigma_{\{\overline{\mathcal{A}}_i, \overline{\mathcal{A}}_j\}}(\overline{\mathcal{A}}_i, \mathbb{D})$ between any two features $\overline{\mathcal{A}}_i$ and $\overline{\mathcal{A}}_j$ of reduced feature set $\mathbb{S}$ is high.
3. If $\lambda_j = 0$ for all $\mathcal{A}_j \in \mathrm{D}_i$, then the extracted feature $\overline{\mathcal{A}}_i$ at a particular iteration is actually the candidate feature $\mathcal{A}_i$ of the original feature set $\mathbb{C}$.

Hence, the proposed dimensionality reduction method generates a reduced feature set $\mathbb{S}$ that may simultaneously contain some selected features of original measurement space and some extracted features of transformed feature space, which are both relevant and significant.

5.2.4 Complexity of the Algorithm

The proposed FR-SFSFEx algorithm has low computational complexity with respect to the number of features present in original set.

The computation of the relevance of m features is carried out in step 2 of the proposed FR-SFSFEx algorithm, which has $\mathcal{O}(m)$ time complexity. The selection of most relevant feature from the set of m features, which is carried out in step 3, has also a complexity $\mathcal{O}(m)$. There is only one loop in step 4 of the proposed dimensionality reduction method, which is executed $(d-1)$ times, where d represents the number of features in the reduced feature set. To generate insignificant, dispensable, and significant sets, the significance of all existing features needs to be computed. The computation of significance of a feature with respect to another feature takes only a constant amount of time. If $\acute{m} < m$ represents the cardinality of the existing feature set, the complexity to compute the significance of $\acute{m}$ features, which is carried out in step 4-a), is $\mathcal{O}(\acute{m})$. The computational complexity of extracting a new feature from the dispensable feature set with cardinality $|\mathrm{D}_i| < \acute{m}$, which is carried out in steps 4-b) and 4-c), is given by $\mathcal{O}(|\mathrm{D}_i|^2)$. The computation of the significance of $(\acute{m} - |\mathrm{D}_i| - |\mathrm{I}_i|)$ candidate features with respect to the already-selected features, which is carried out in step 4-e), has a complexity $\mathcal{O}(\acute{m} - |\mathrm{D}_i| - |\mathrm{I}_i|)$, where $|\mathrm{I}_i|$ is the cardinality of the insignificant set. In effect, the selection of a feature from $(\acute{m} - |\mathrm{D}_i| - |\mathrm{I}_i|)$ candidate features by maximizing both relevance and significance has also a complexity $\mathcal{O}(\acute{m} - |\mathrm{D}_i| - |\mathrm{I}_i|)$.

Hence, the total complexity to execute the loop $(d-1)$ times is $\mathcal{O}((d-1)(\acute{m}+|\mathrm{D}_i|^2+(\acute{m}-|\mathrm{D}_i|-|\mathrm{I}_i|)))$. In effect, the selection of a set of d features from the whole set of m features using the proposed FR-SFSFEx algorithm has an overall computational complexity of $\mathcal{O}(m)+\mathcal{O}(m)+\mathcal{O}((d-1)(\acute{m}+|\mathrm{D}_i|^2+(\acute{m}-|\mathrm{D}_i|-|\mathrm{I}_i|))) \simeq \mathcal{O}(m+(d-1)(\acute{m}+|\mathrm{D}_i|^2-|\mathrm{I}_i|))$, where $|\mathrm{D}_i|, |\mathrm{I}_i| < \acute{m} < m$.

5.2.5 Selection of Threshold

The threshold δ_i in (5.2) plays an important role to form the dispensable set corresponding to the candidate feature $\mathcal{A}_i$ at a particular iteration. It controls the degree of similarity among the features in the dispensable set. In effect, it has a direct influence on the performance of proposed FR-SFSFEx algorithm. If δ_i increases, the number of features or attributes in the dispensable set increases, but the similarity among them with respect to

sample categories or class labels decreases. The similarity among the features in dispensable set increases with a decrease of δ_i.

To find out the optimum value of δ_i, the following definition, based on the significance values of the candidate feature set $\mathcal{B}$ for each iteration, can be used:

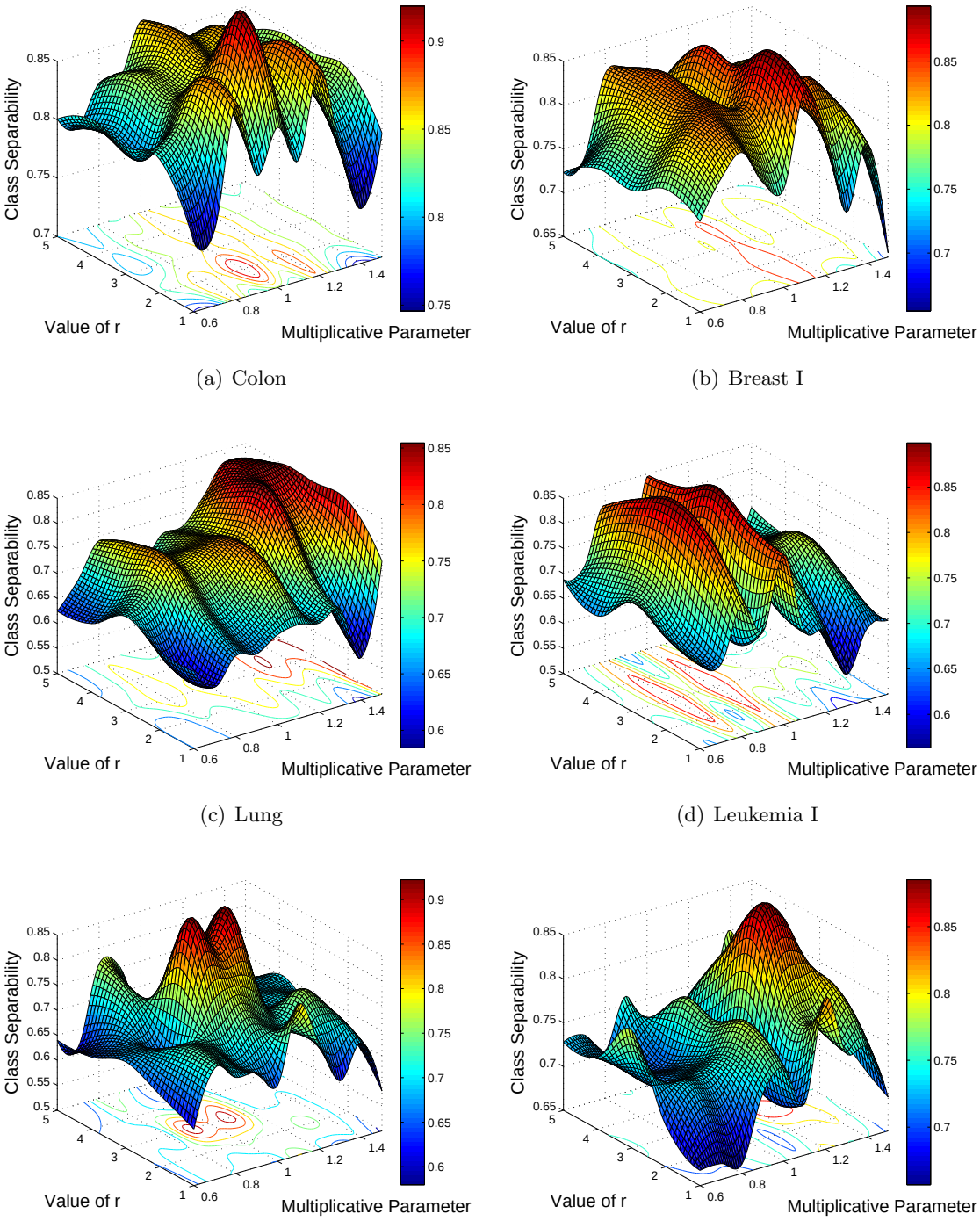
$$\delta_i = \left[\frac{1}{|\mathcal{B}|} \sum_{\mathcal{A}_j \in \mathcal{B}} \left\{ \sigma_{\{\mathcal{A}_i, \mathcal{A}_j\}}(\mathcal{A}_j, \mathbb{D}) \right\}^r \right]^{\frac{1}{r}}; \tag{5.9}$$

where r is a positive integer. Hence, the threshold δ_i represents the zero mean, rth order moment of significance values of the attributes $\mathcal{A}_j \in \mathcal{B}$ with respect to the candidate feature $\mathcal{A}_i$.

5.3 Experimental Results and Discussion

The performance of the proposed FR-SFSFEx method is extensively studied and compared with that of some existing feature selection and extraction algorithms, namely, maximal-relevance (Max-Relevance) and maximal-relevance maximal-significance (MRMS) [327] frameworks with classical, neighborhood, and fuzzy-rough sets, quick reduct (Max-Dependency and rough sets) [83], fuzzy-rough quick reduct (Max-Dependency and fuzzy-rough sets) [220], neighborhood quick reduct (Max-Dependency and neighborhood rough sets) [198], minimal-redundancy maximal-relevance (mRMR) framework [416], fuzzy-rough set based mRMR framework (fuzzy-rough mRMR) [323], margin based approaches such as relevance in estimating features (RELIEF) [252], iterative search margin based algorithm (SIMBA) [152]; wrapper approach (Wrapper), wrapper with forward search (WFS), combinatorial correlation-based feature selection algorithm (CFS) [173]; and existing feature extraction algorithms, namely, principal component analysis (PCA) [112], independent component analysis (ICA) [112,498], linear discriminant analysis (LDA) [112], and partial least squares discriminant analysis (PLS-DA) [511]. Ten benchmark data sets are used for the comparison purpose. While Satimage, Segmentation, Isolet, and Multiple Features data sets are downloaded from the *UCI Machine Learning Repository* [134], Breast Cancer I, Breast Cancer II, Colon Cancer, Lung Cancer, Leukemia I, and Leukemia II data sets are available at the *Kent Ridge Bio-medical Data Set Repository* [1]. The descriptions of these data sets are given in Section 3.3.2. All the algorithms are implemented in C language and run in Ubuntu 11.04 environment with 64 bit support having machine configuration of Pentium Core 2 Quad 2.66 GHz with 4 MB L2 cache, and 4 GB DDR2 RAM.

Three pattern classifiers, namely, K-NN, SVM, and C4.5, are used to evaluate the performance of different dimensionality reduction methods with respect to several real life data sets, which are described in Section 3.3.2 of Chapter 3. To compute the classification



(a) Colon

(b) Breast I

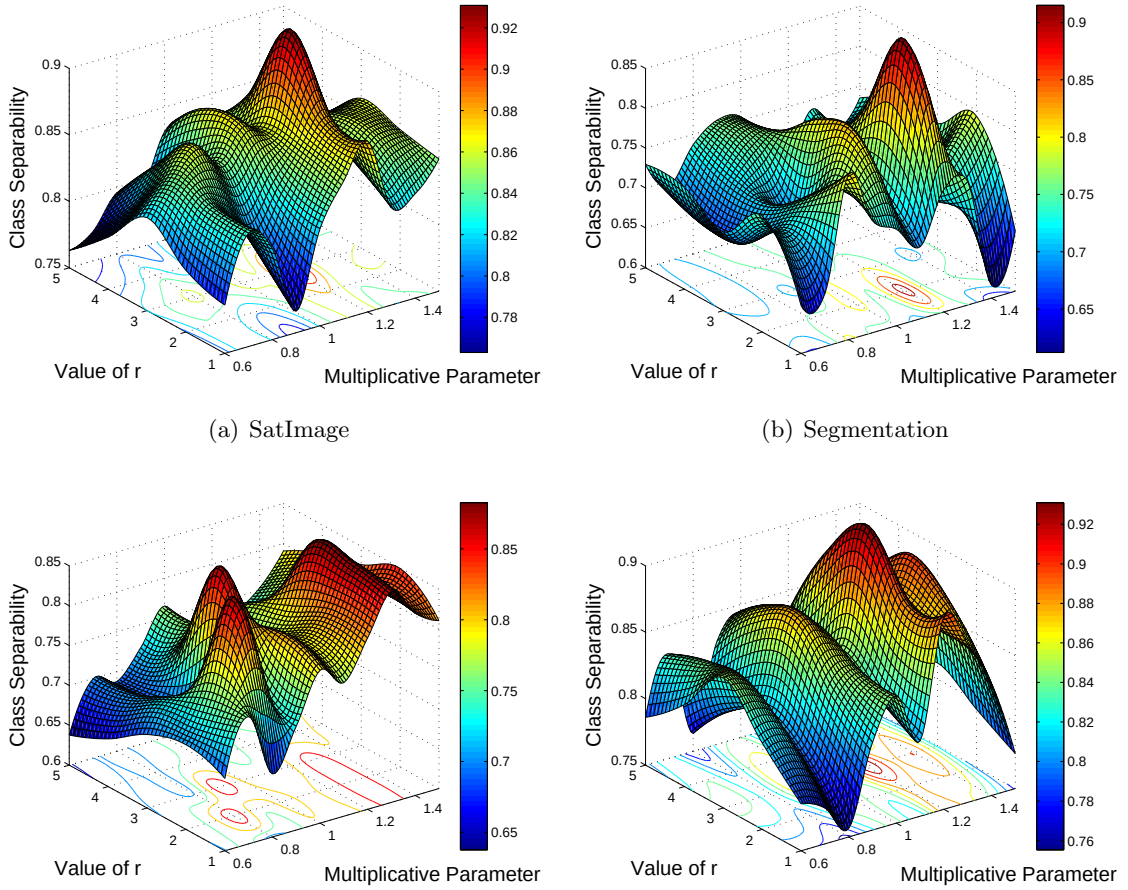
(c) Lung

(d) Leukemia I

(e) Isolet

(f) Multiple Features

Figure 5.1: Variation of class separability index for different values of moment order r and multiplicative parameter η



(a) SatImage (b) Segmentation

(c) Leukemia II (d) Breast II

Figure 5.2: Variation of class separability index for different values of moment order r and multiplicative parameter η

accuracy of the K-NN rule, SVM, and C4.5, training-testing, 10 fold cross-validation (CV), and bootstrap $\mathcal{E}_{\text{BS}}^{632}$ (Appendix A) are performed. The 10-fold CV and bootstrap $\mathcal{E}_{\text{BS}}^{632}$ are performed on Colon Cancer, Breast Cancer I, Lung Cancer, Leukemia I, Isolet, and Multiple Features data sets, while the training-testing is done on Satimage, Segmentation, Leukemia II, and Breast Cancer II data sets. Some of the results of this chapter have also been reported in [314].

5.3.1 Optimum Values of Different Parameters

The multiplicative parameter η in (3.11) of Chapter 3 of fuzzy-rough sets controls the extent of overlapping between the fuzzy equivalence classes low and medium or medium and high. Keeping the values of σ_{low} and σ_{high} in (3.10) of Chapter 3 fixed, the amount of overlapping among the three π functions can be altered varying σ_{medium} ((3.11) of

Chapter 3). As η is decreased, the radius σ_{medium} decreases around $\bar{c}_{\text{medium}}$ such that ultimately there is insignificant overlapping between the π functions low and medium or medium and high. On the other hand, as η is increased, the radius σ_{medium} increases around $\bar{c}_{\text{medium}}$ so that the amount of overlapping between π functions increases.

The parameter r in (5.9) is the moment order of significance values of all features present at a particular iteration. In effect, it controls the size of the insignificant, dispensable, and significant feature sets corresponding to the candidate feature of that iteration. Hence, the quality of the extracted features at various iterations as well as the overall performance of the proposed dimensionality reduction method very much depend on the value of r. Let $\mathrm{S} = \{r, \eta\}$ be the set of parameters and $\mathrm{S}^* = \{r^\star, \eta^\star\}$ is the set of optimal parameters. To find out the optimum values of r and η for a given data set, the class separability index [101] is used, which is described in Section 4.4.1 of Chapter 4.

For all data sets, the values of moment order r and multiplicative parameter η vary from 1 to 5 and 0.6 to 1.5, respectively. Fig. 5.1 and 5.2 represent the variation of class separability index $\mathcal{S}$ for fuzzy-rough sets with respect to different values of r and η. The results are reported for all the data sets. In case of Satimage, Segmentation, Leukemia II, and Breast II, the average class separability index values are presented for different folds of training samples. From the results reported in Fig. 5.1 and 5.2, it is seen that, as the values of both r and η increase, the class separability index $\mathcal{S}$ also increases and attains its maximum value at the particular values of $r^\star$ and $\eta^\star$. Hence, the optimum values of $r^\star$ and $\eta^\star$ are obtained using the following relation:

$$\mathrm{S}^\star = \arg\max_{\mathrm{S}} \{\mathcal{S}\}. \tag{5.10}$$

Table 5.1 presents the optimum values of $\{r, \eta\}$ for fuzzy-rough sets obtained using (5.10), along with the optimum values of r for classical rough sets and $\{r, \Phi\}$ for neighborhood rough sets, on different data sets. From the results reported in Table 5.1, it can be seen that, in case of proposed fuzzy-rough dimensionality reduction method, the optimum value of r is either 2 or 3 in most of the cases, while η varies between 1.0 and 1.5. For class separability index, DB index, Dunn index, and FFE index, $r = 3$ is used.

5.3.2 Effectiveness of Parameter Optimization Technique

In order to establish the effectiveness of proposed method for finding optimum values of different parameters, extensive experimentation is done on different real life data sets. To compute the relevance and significance of the feature set in the proposed dimensionality reduction method, classical or Pawlak's rough sets, neighborhood rough sets, fuzzy-rough sets, class separability index, DB index, Dunn index, and FFE index are used.

Table 5.2 presents the performance of classical, neighborhood, and fuzzy-rough sets on

Table 5.1: Optimum Values of Parameters for Classical, Neighborhood, and Fuzzy-Rough based SFSFEx on Different Data Sets

Experimental Setup	Different Data Sets	Different Rough Sets		
		Classical: $S^\star = r^\star$	Neighborhood: $S^\star = \{r^\star, \Phi^\star\}$	Fuzzy: $S^\star = \{r^\star, \eta^\star\}$
Training/ Testing	Satimage	2	{2, 0.40}	{3, 1.2}
	Segmentation	2	{2, 0.05}	{2, 1.2}
	Leukemia II	4	{2, 0.25}	{2, 0.8/1.3}/{3, 0.9/1.3}
	Breast Cancer II	4/5	{2, 0.20/0.25}	{3, 1.2}
10-fold Cross Validation	Colon Cancer	2	{3, 0.20}	{2, 0.9/1.1}
	Breast Cancer I	3	{4, 0.30/0.35}	{2, 1.2}
	Lung Cancer	3	{3, 0.20/0.25}	{3, 1.3/1.4}
	Leukemia I	2	{2, 0.25}/{3, 0.20}	{2, 0.8/1.0}/{3,1.9/1.0}
	Isolet	2	{3, 0.35}	{3, 0.9/1.1}
	Multiple Features	3	{3, 0.25/0.30}	{3, 1.2/1.3}
Bootstrap $\mathcal{E}_{\text{BS}}^{632}$	Colon Cancer	2	{3/4, 0.25/0.30}	{2, 1.0}/{3, 1.1}
	Breast Cancer I	3	{3, 0.20/0.25}/{4, 0.20}	{3, 1.5}
	Lung Cancer	2	{3, 0.20/0.25}	{2, 1.4/1.5}/{3, 1.3/1.4}
	Leukemia I	3	{3/4, 0.10/0.20}	{3, 1.0/1.5}
	Isolet	3	{4, 0.25/0.30}	{2, 0.8}/{4, 1.2/1.3}
	Multiple Features	3	{3, 0.15/0.20}/{4, 0.20}	{3, 1.2/1.3}

Table 5.2: Classification Accuracy of SFSFEx for Optimum Parameters on Different Data Sets

SFSFEx Models	Different Data Sets	Test Accuracy of K-NN			Test Accuracy of SVM			Test Accuracy of C4.5		
		Best	Best Training	Proposed	Best	Best Training	Proposed	Best	Best Training	Proposed
Classical Rough Sets	Satimage	84.7/18	84.7/28	**84.7/18**	84.5/13	82.7/17	**84.5/13**	**85.9/23**	82.4/22	82.4/17
	Segmentation	87.6/21	87.6/21	**87.6/24**	91.4/10	91.4/10	**91.4/10**	90.9/13	90.9/22	**90.9/19**
	Leukemia II	91.1/28	91.1/28	**91.1/28**	91.1/13	91.1/19	**91.1/13**	91.1/19	91.1/26	**91.1/19**
	Breast II	**89.5/18**	89.5/23	84.2/17	**94.7/18**	94.7/18	89.5/24	100.0/9	100.0/11	**100.0/13**
	Colon	**75.2/23**	75.2/23	73.4/15	71.8/21	71.6/26	**71.8/25**	**80.3/22**	78.4/8	78.4/14
	Breast I	73.9/22	73.9/28	**73.9/22**	**71.0/13**	70.6/12	70.6/11	72.3/20	70.8/20	**72.3/22**
	Lung	74.2/17	73.4/28	**74.2/17**	**74.5/27**	71.7/16	73.4/18	75.6/25	75.6/25	**75.6/26**
	Leukemia I	**78.0/12**	75.0/17	75.6/26	75.7/23	74.0/15	**75.7/24**	**77.0/21**	75.2/24	75.2/9
	Isolet	73.1/8	73.1/28	**73.1/28**	80.1/21	80.1/21	**80.1/21**	63.3/14	62.0/13	**63.3/23**
	Mult. Feat.	81.0/17	81.0/17	**81.0/22**	79.0/21	79.0/27	**79.0/27**	75.0/10	75.0/10	**75.0/10**
Neighborhood Rough Sets	Satimage	87.5/22	87.5/22	**87.5/25**	83.6/8	83.6/18	**83.6/10**	**86.0/24**	84.7/8	84.0/15
	Segmentation	85.8/17	85.8/24	**85.8/17**	91.4/13	91.4/19	**91.4/19**	90.0/16	90.0/25	**90.0/16**
	Leukemia II	**93.8/13**	93.8/13	91.1/25	**93.8/21**	93.8/23	90.2/10	**91.1/9**	91.1/9	90.2/16
	Breast II	94.7/15	94.7/23	**94.7/21**	94.7/11	94.7/24	**94.7/18**	100.0/12	100.0/24	**100.0/24**
	Colon	79.6/19	79.6/25	**79.6/19**	**80.9/24**	77.7/18	79.3/12	78.2/19	76.7/9	**78.2/26**
	Breast I	74.9/16	73.4/9	**74.9/16**	71.9/10	71.9/10	**71.9/10**	**80.7/12**	78.7/25	78.7/18
	Lung	**79.6/15**	76.1/26	77.6/15	75.7/11	74.2/28	**75.7/11**	75.9/12	74.4/12	**75.9/12**
	Leukemia I	76.6/17	76.6/25	**76.6/17**	**81.6/21**	80.0/17	80.0/16	79.8/13	79.8/17	**79.8/13**
	Isolet	68.2/13	68.2/21	**68.2/15**	77.3/22	77.3/22	**77.3/27**	72.3/18	70.8/16	**72.3/19**
	Mult. Feat.	**85.5/25**	82.2/18	83.8/10	86.1/14	84.4/15	**86.1/14**	**88.7/21**	85.2/25	87.0/10
Fuzzy Rough Sets	Satimage	**88.0/25**	87.6/20	87.6/8	87.4/13	87.4/24	**87.4/13**	**87.4/11**	87.4/13	87.2/11
	Segmentation	**86.3/12**	86.2/12	86.2/14	92.3/10	92.3/14	**92.3/10**	90.3/14	90.3/17	**90.3/15**
	Leukemia II	94.6/9	94.6/9	**94.6/9**	93.8/13	93.8/13	**93.8/13**	93.8/8	93.8/16	**93.8/8**
	Breast II	94.7/12	94.7/14	**94.7/12**	94.7/11	94.7/23	**94.7/12**	100.0/10	100.0/17	**100.0/10**
	Colon	79.5/9	77.9/24	**79.5/9**	80.4/10	78.8/18	**80.4/10**	78.9/13	78.9/13	**78.9/14**
	Breast I	**80.8/15**	76.3/13	76.3/13	**74.1/18**	72.7/20	72.7/13	74.4/10	72.9/16	**74.4/10**
	Lung	85.1/12	85.1/12	**85.1/12**	83.2/10	83.2/27	**83.2/10**	87.5/9	85.7/14	**87.5/9**
	Leukemia I	85.4/11	83.7/19	**85.4/12**	78.6/12	77.0/23	**78.6/13**	85.7/8	85.7/27	**85.7/8**
	Isolet	85.4/16	83.7/12	**85.4/16**	85.9/8	85.9/22	**85.9/8**	80.8/11	80.8/21	**80.8/13**
	Mult. Feat.	91.6/12	91.6/15	**91.6/12**	94.7/8	94.7/9	**94.7/9**	91.8/12	91.8/15	**91.8/12**

Table 5.3: Classification Accuracy of SFSFEx for Optimum Parameters on Different Data Sets

SFSFEx Models	Different Data Sets	Test Accuracy of K-NN			Test Accuracy of SVM			Test Accuracy of C4.5		
		Best	Best Training	Proposed	Best	Best Training	Proposed	Best	Best Training	Proposed
Class Separability Index	Satimage	82.7/15	82.7/15	**82.7/15**	**83.7/27**	82.4/13	82.9/10	**84.2/9**	83.1/14	83.1/11
	Segmentation	78.9/16	78.9/26	**78.9/16**	71.2/19	71.2/19	**71.2/19**	80.7/11	80.7/22	**80.7/11**
	Leukemia II	**83.9/18**	83.9/24	83.0/21	**83.9/22**	83.0/14	83.0/27	83.0/13	83.0/18	**83.0/13**
	Breast II	**84.2/8**	84.2/28	78.9/19	**84.2/17**	79.8/16	78.9/16	100.0/13	100.0/13	**100.0/22**
	Colon	76.3/19	74.8/28	**76.3/19**	73.9/9	72.4/11	**73.9/21**	77.1/10	75.5/23	**77.1/14**
	Breast I	73.7/13	72.2/23	**73.7/13**	**73.0/19**	69.8/9	71.2/18	70.2/10	70.2/23	**70.2/24**
	Lung	60.0/15	58.8/8	**60.0/26**	58.8/20	58.8/25	**58.8/22**	71.7/20	70.3/27	**71.7/22**
	Leukemia I	77.9/10	77.9/10	**77.9/10**	**80.9/8**	79.3/21	79.3/16	**78.1/12**	76.5/26	76.5/12
	Isolet	47.2/13	47.2/13	**47.2/13**	47.1/13	47.1/18	**47.1/16**	44.5/21	44.5/21	**44.5/29**
	Mult. Feat.	**75.5/19**	73.6/12	73.6/15	69.5/19	68.6/24	**69.5/19**	73.0/12	71.6/16	**73.0/16**
DB Index	Satimage	**85.3/19**	84.7/15	84.7/27	85.6/11	85.6/29	**85.6/20**	**86.1/16**	85.3/10	85.3/10
	Segmentation	78.6/18	78.6/18	**78.6/18**	91.2/18	91.2/21	**91.2/20**	**90.0/14**	88.6/14	87.5/18
	Leukemia II	97.5/13	97.5/22	**97.5/13**	**93.8/17**	88.4/17	88.4/22	**89.3/23**	88.4/8	88.4/26
	Breast II	94.7/8	94.7/24	**94.7/12**	94.7/14	94.7/25	**94.7/23**	100.0/9	100.0/26	**100.0/17**
	Colon	74.7/17	74.7/28	**74.7/19**	74.9/9	74.9/11	**74.9/23**	79.5/16	77.9/21	**79.5/22**
	Breast I	76.4/13	74.9/20	**76.4/22**	70.8/23	70.8/24	**70.8/23**	**73.6/8**	72.7/24	71.8/21
	Lung	64.8/11	64.8/28	**64.8/11**	**60.8/21**	59.3/15	59.3/12	60.2/14	59.0/15	**60.2/22**
	Leukemia I	**81.1/25**	79.1/10	79.1/28	76.2/10	74.7/11	**76.2/13**	**80.6/22**	79.0/9	79.0/22
	Isolet	67.0/20	67.0/20	**67.0/20**	68.7/15	67.3/22	**68.7/16**	65.6/10	65.6/14	**65.6/20**
	Mult. Feat.	74.5/15	73.0/9	**74.5/17**	**74.6/25**	73.1/19	73.1/19	74.2/21	74.2/27	**74.2/26**
Dunn Index	Satimage	**85.3/23**	84.5/9	84.5/17	**84.8/21**	84.4/27	84.4/28	**86.0/13**	85.3/10	85.3/10
	Segmentation	78.4/15	78.4/27	**78.4/25**	90.3/10	90.3/18	**90.3/10**	87.5/14	87.5/14	**87.5/25**
	Leukemia II	**91.1/24**	85.7/12	85.7/24	93.8/15	93.8/19	**93.8/15**	**91.1/22**	88.9/17	89.3/8
	Breast II	94.7/19	94.7/23	**94.7/23**	94.7/12	94.7/14	**94.7/21**	100.0/10	100.0/10	**100.0/17**
	Colon	74.6/14	74.6/26	**74.6/20**	**76.8/13**	75.3/15	75.3/26	75.0/11	73.5/16	**75.0/16**
	Breast I	75.2/9	75.2/10	**75.2/10**	70.6/9	70.6/16	**70.6/9**	71.2/9	71.2/19	**71.2/22**
	Lung	**65.7/23**	62.8/13	64.1/15	74.7/11	74.7/18	**74.7/11**	73.7/10	72.3/10	**73.7/21**
	Leukemia I	78.7/15	78.7/22	**78.7/15**	**76.9/14**	75.6/20	75.1/26	**80.0/16**	78.1/25	78.1/18
	Isolet	65.8/13	65.8/15	**65.8/13**	67.6/12	66.2/20	**67.6/13**	65.1/17	65.1/25	**65.1/20**
	Mult. Feat.	**76.0/21**	74.2/21	74.2/15	73.2/14	73.2/18	**73.2/26**	73.8/10	73.8/10	**73.8/10**
FFE Index	Satimage	80.9/20	80.9/26	**80.9/23**	**81.8/20**	80.7/19	80.7/10	80.2/13	80.2/18	**80.2/14**
	Segmentation	85.8/15	85.8/28	**85.8/15**	91.3/11	91.3/11	**91.3/11**	85.5/12	85.5/12	**85.5/15**
	Leukemia II	**83.9/25**	83.0/19	83.0/27	89.3/18	89.3/22	**89.3/18**	**89.3/12**	89.3/26	87.5/12
	Breast II	84.2/22	84.2/27	**84.2/27**	84.2/13	84.2/13	**84.2/27**	87.2/23	86.8/25	**87.2/23**
	Colon	68.5/18	67.1/25	**68.5/22**	**71.0/20**	68.2/18	69.6/11	**69.8/10**	67.0/25	68.4/20
	Breast I	67.8/9	67.8/9	**67.8/9**	61.9/12	60.7/22	**61.9/12**	**62.8/25**	60.0/20	61.3/9
	Lung	**69.3/21**	67.6/10	67.6/24	65.9/10	65.9/10	**65.9/24**	65.7/11	65.7/11	**65.7/14**
	Leukemia I	**70.1/11**	68.4/25	68.4/12	69.0/14	69.0/15	**69.0/14**	**70.6/11**	69.0/18	68.8/21
	Isolet	64.3/15	64.3/21	**64.3/21**	**69.0/14**	61.6/13	62.9/12	60.9/24	60.9/26	**60.9/29**
	Mult. Feat.	70.2/9	70.2/9	**70.2/9**	**68.2/19**	66.9/16	66.9/13	68.0/15	68.0/24	**68.0/15**

Table 5.4: Comparative Performance of Various SFSFEx Models on Different Data Sets with respect to K-NN Classifier

Exp Setup	Different Data Sets	Different Statistics	Class Separability	DB Index	Dunn Index	FFE Index	Classical Rough	Neighborhood Rough	Fuzzy Rough
Training/ Testing	Satimage	Classification Accuracy	82.7/23	84.7/9	84.5/25	80.9/28	84.7/26	87.5/12	**87.6/8**
	Segmentation		78.9/28	78.6/13	78.4/17	85.8/12	**87.6**/22	85.8/18	86.2/14
	Leukemia II		83.0/18	97.5/19	85.7/19	83.0/11	91.1/8	91.1/21	**94.6/9**
	Breast II		78.9/25	94.7/24	94.7/20	84.2/15	84.2/25	94.7/22	**94.7/12**
10-fold Cross Validation	Colon	Mean	76.3/23	74.7/12	74.6/19	68.5/12	73.4/11	79.6/23	79.5/9
		StdDev	6.66	5.97	6.29	6.74	6.45	7.61	6.74
		Comp	1.06	1.69	1.66	**3.64**	**2.07**	-0.03	
	Breast I	Mean	73.7/26	76.4/9	75.2/19	67.8/18	73.9/17	74.9/24	76.3/13
		StdDev	7.97	7.64	6.01	7.37	5.05	6.53	7.54
		Comp	0.74	-0.04	0.36	**2.53**	0.81	0.44	
	Lung	Mean	60.0/23	64.8/11	64.1/15	67.6/8	74.2/17	77.6/12	85.1/12
		StdDev	5.23	7.20	5.70	5.33	5.66	5.09	6.36
		Comp	**9.66**	**6.68**	**7.78**	**6.67**	**4.05**	**2.9**	
	Leukemia I	Mean	77.9/19	79.1/17	78.7/20	68.4/13	75.6/9	76.6/23	85.4/12
		StdDev	5.16	7.23	5.00	6.34	5.56	5.15	8.94
		Comp	**2.30**	1.72	**2.05**	**4.89**	**2.95**	**2.68**	
	Isolet	Mean	47.2/16	67.0/10	65.8/11	64.3/15	73.1/20	68.2/21	85.4/16
		StdDev	5.82	5.71	5.87	6.25	7.65	7.91	6.88
		Comp	**13.40**	**6.51**	**6.83**	**7.18**	**3.76**	**5.19**	
	Multiple Features	Mean	73.6/9	74.5/17	74.2/17	70.2/26	81.0/25	83.8/10	91.6/12
		StdDev	7.00	7.00	6.23	5.42	5.97	5.31	5.73
		Comp	**6.29**	**5.99**	**6.53**	**8.61**	**4.05**	**3.16**	
Bootstrap $\mathcal{E}_{BS}^{632}$	Colon		72.9/31	71.2/26	70.7/23	65.8/35	86.7/24	83.4/31	**88.5/37**
	Breast I		67.0/20	69.6/34	70.3/21	66.4/27	81.1/37	83.3/37	**81.6/37**
	Lung		68.0/26	71.6/32	75.1/24	78.6/36	80.5/21	80.4/41	**90.8/43**
	Leukemia I		70.8/28	71.6/31	70.1/32	67.7/36	82.3/20	84.2/32	**91.7/41**
	Isolet		57.3/34	70.3/33	**83.2/30**	76.8/35	78.1/25	80.8/42	82.7/37
	Multiple Features		66.7/29	67.1/34	67.8/28	65.6/36	74.6/36	81.5/45	**86.5/44**

Table 5.5: Comparative Performance of Various SFSFEx Models on Different Data Sets with respect to SVM Classifier

Exp Setup	Different Data Sets	Different Statistics	Class Separability	DB Index	Dunn Index	FFE Index	Classical Rough	Neighborhood Rough	Fuzzy Rough
Training/ Testing	Satimage	Classification Accuracy	82.9/19	85.6/18	84.4/21	80.7/16	84.5/16	83.6/18	**87.4/13**
	Segmentation		71.2/13	91.2/24	90.3/18	91.3/11	91.4/8	91.4/13	**92.3/10**
	Leukemia II		83.0/20	88.4/13	93.8/27	89.3/19	91.1/25	90.2/26	**93.8/13**
	Breast II		78.9/18	94.7/24	94.7/26	84.2/24	89.5/10	94.7/11	**94.7/12**
10-fold Cross Validation	Colon	Mean	73.9/23	74.9/18	75.3/14	69.6/21	71.8/25	79.3/21	80.4/10
		StdDev	7.33	5.84	6.53	6.00	6.13	6.98	6.88
		Comp	**2.04**	**1.92**	1.70	**3.73**	**2.94**	0.34	
	Breast I	Mean	71.2/10	70.8/27	70.6/17	61.9/9	70.6/25	71.9/8	72.7/13
		StdDev	7.17	5.31	6.90	6.63	5.24	6.32	7.84
		Comp	0.43	0.62	0.61	**3.30**	0.70	0.24	
	Lung	Mean	58.8/23	59.3/13	74.7/12	65.9/23	73.4/19	75.7/19	83.2/10
		StdDev	6.77	5.97	6.37	5.32	5.66	7.16	8.33
		Comp	**7.20**	**7.38**	**2.57**	**5.54**	**3.09**	**2.17**	
	Leukemia I	Mean	79.3/23	76.2/20	75.1/23	69.0//9	75.7/21	80.0/8	78.6/13
		StdDev	7.08	5.47	6.78	5.50	5.52	5.37	8.76
		Comp	-0.22	0.72	0.99	**2.91**	0.86	-0.43	
	Isolet	Mean	47.1/18	68.7/20	67.6/15	62.9/21	80.1/12	77.3/21	85.9/8
		StdDev	7.65	6.74	6.26	5.87	6.08	5.22	8.56
		Comp	**10.68**	**4.99**	**5.46**	**7.01**	1.75	**2.71**	
	Multiple Features	Mean	69.5/19	73.1/26	73.2/18	66.9//9	79.0/12	86.1/18	94.7/9
		StdDev	7.79	6.32	7.30	5.13	6.81	6.63	3.88
		Comp	**9.15**	**9.23**	**8.23**	**13.70**	**6.34**	**3.56**	
Bootstrap $\mathcal{E}_{BS}^{632}$	Colon		67.4/31	69.7/36	75.6/37	66.5/24	71.4/37	71.9/39	**72.4/39**
	Breast I		67.7/30	66.7/29	74.2/24	62.8/20	66.0/26	72.1/30	**77.2/34**
	Lung		73.3/36	65.8/28	81.3/23	78.3/40	72.6/29	75.0/33	**84.0/40**
	Leukemia I		72.4/38	73.4/33	71.5/35	63.4/38	72.6/41	73.3/32	**74.5/46**
	Isolet		62.7/22	76.8/35	73.3/24	69.9/37	76.0/27	79.7/43	**84.6/39**
	Multiple Features		66.9/31	70.9/34	71.2/39	63.7/35	82.4/45	86.4/45	**90.7/33**

Table 5.6: Comparative Performance of Various SFSFEx Models on Different Data Sets with respect to C4.5 Classifier

Exp Setup	Different Data Sets	Different Statistics	Class Separability	DB Index	Dunn Index	FFE Index	Classical Rough	Neighborhood Rough	Fuzzy Rough
Training/ Testing	Satimage	Classification Accuracy	83.1/15	85.3/16	85.3/24	80.2/13	82.4/12	84.0/10	**87.2/11**
	Segmentation		80.7/18	87.5/25	87.5/14	85.5/14	90.9/16	90.0/22	90.3/15
	Leukemia II		83.0/28	88.4/25	89.3/19	87.5/26	91.1/14	90.2/13	**93.8/8**
	Breast II		100.0/11	100.0/24	100.0/24	87.2/26	100.0/20	100.0/14	**100.0/10**
10-fold Cross Validation	Colon	Mean	77.1/12	79.5/19	75.0/11	68.4/19	78.4/22	78.2/8	78.9/14
		StdDev	5.96	6.37	6.49	5.58	7.89	6.67	9.48
		Comp	0.52	-0.15	1.10	**3.03**	0.15	0.19	
	Breast I	Mean	70.2/18	71.8/9	71.2/24	61.3/9	72.3/19	78.7/15	74.4/10
		StdDev	5.46	7.22	6.10	7.35	6.60	6.69	10.22
		Comp	1.14	0.64	0.85	**3.30**	0.55	-1.13	
	Lung	Mean	71.7/23	60.2/27	73.7/17	65.7/11	75.6/23	75.9/8	87.5/9
		StdDev	6.79	6.25	6.18	7.69	5.10	6.11	6.88
		Comp	**5.15**	**9.27**	**4.70**	**6.68**	**4.38**	**3.97**	
	Leukemia I	Mean	76.5/17	79.0/10	78.1/28	68.8/11	75.2/20	79.8/11	85.7/8
		StdDev	5.22	7.17	6.82	5.15	7.86	7.75	8.96
		Comp	**2.78**	**1.83**	**2.13**	**5.15**	**2.79**	1.56	
	Isolet	Mean	44.5/18	65.6/17	65.1/15	60.9/21	63.3/27	72.3/22	80.8/13
		StdDev	7.55	5.64	7.81	5.42	6.84	5.11	5.87
		Comp	**11.99**	**5.88**	**5.06**	**7.88**	**6.14**	**3.45**	
	Multiple Features	Mean	73.0/24	74.2/12	73.8/19	68.0/11	75.0/19	87.0/15	91.8/12
		StdDev	6.12	5.78	6.41	7.24	7.70	5.09	6.38
		Comp	**6.70**	**6.46**	**6.26**	**7.78**	**5.31**	**1.87**	
Bootstrap $\mathcal{E}_{BS}^{632}$	Colon		70.6/30	69.7/37	69.0/39	69.8/31	72.3/29	79.4/36	**82.7/44**
	Breast I		70.5/38	71.6/30	74.7/28	59.4/23	73.5/35	76.5/44	**84.0/47**
	Lung		70.1/35	65.9/39	75.9/37	72.1/36	76.4/44	74.8/41	**84.5/33**
	Leukemia I		67.9/26	70.2/35	71.7/38	67.9/29	76.6/33	78.0/41	**78.1/37**
	Isolet		58.9/27	74.1/28	71.8/25	72.8/24	72.7/33	74.0/37	**79.5/40**
	Multiple Features		66.8/37	66.2/29	67.3/21	62.7/35	78.1/26	80.5/41	**83.2/39**

Table 5.7: Comparative Performance Analysis of Different Methods on Satimage, Segmentation, Leukemia II, and Breast II

Different	Satimage			Segmentation			Leukemia II			Breast II		
Methods / Algorithms	K-NN	SVM	C4.5	K-NN	SVM	C4.5	K-NN	SVM	C4.5	K-NN	SVM	C4.5
Wrapper	84.8/29	80.8/7	85.1/34	87.9/24	75.5/16	91.5/11	81.5/28	84.9/19	79.3/26	79.3/38	86.2/30	90.6/32
WFS	84.8/17	80.8/39	85.1/23	87.9/36	75.5/32	91.5/38	81.6/32	83.5/18	77.8/30	78.9/13	85.3/21	89.3/39
CFS	84.4/24	82.2/29	85.1/16	88.5/22	57.4/38	86.2/10	79.4/13	82.3/32	80.4/27	80.3/9	87.2/29	88.6/32
PLS-DA	82.7/30	82.7/28	82.7/18	84.5/20	85.2/23	91.1/23	81.3/20	79.0/25	81.6/24	79.3/35	87.2/18	89.4/28
FR-SFSFEx	**87.6/8**	**87.4/13**	**87.2/11**	**86.2/14**	**92.3/10**	90.3/15	**94.6/9**	**93.8/13**	**93.8/8**	**94.7/12**	**94.7/12**	**100.0/10**

Table 5.8: Bootstrap $\mathcal{E}_{\mathrm{BS}}^{.632}$ Analysis of Different Methods on Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features

Different	Different	Different Data Sets					
Classifiers	Methods / Algorithms	Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	Wrapper	80.5/22	76.3/20	72.0/23	84.5/15	67.5/19	73.2/23
	WFS	72.3/15	68.7/28	79.9/13	78.9/20	78.8/20	83.2/20
	CFS	75.7/13	80.0/14	79.4/16	88.2/21	74.5/11	81.7/17
	PLS-DA	75.0/18	80.2/22	80.5/9	80.6/16	74.3/10	83.6/11
	FR-SFSFEx	**88.5/37**	**81.6/37**	**90.8/43**	**91.7/41**	**82.7/37**	**86.5/44**
SVM	Wrapper	66.5/13	72.9/12	77.1/21	67.7/18	71.5/15	75.0/14
	WFS	62.7/11	69.0/15	69.2/24	68.5/15	69.2/22	70.2/25
	CFS	66.0/17	68.6/19	72.7/20	71.4/17	76.3/14	77.9/26
	PLS-DA	71.0/28	76.4/19	78.4/11	74.5/10	73.4/13	81.7/12
	FR-SFSFEx	**72.4/39**	**77.2/34**	**84.0/40**	**74.5/46**	**84.6/39**	**90.7/33**
C4.5	Wrapper	76.2/17	80.5/29	76.1/23	72.1/25	57.9/27	76.7/11
	WFS	78.3/25	78.5/27	75.6/22	72.6/24	55.3/12	79.2/14
	CFS	75.4/24	74.6/18	81.1/26	76.6/23	72.2/14	81.5/17
	PLS-DA	75.9/14	87.8/16	76.0/27	83.3/26	67.1/10	81.8/29
	FR-SFSFEx	**82.7/44**	**84.0/47**	**84.5/33**	**78.1/37**	79.5/40	**83.2/39**

Table 5.9: Statistical Significance Test of 10-fold CV with Different Rough Sets on Colon, Breast I, Lung, Leuke I, Isolet, and MFeat

Different Classifiers	Different Criteria	Different Rough Sets	Different Data Sets					
			Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	Max Relevance	Classical	**4.754287**	**4.720209**	**3.954493**	**4.522734**	**5.761566**	**5.211426**
		Neighborhood	**4.780890**	**2.724071**	**2.910541**	**2.299456**	**3.347047**	**4.732598**
		T1 Fuzzy	**2.542935**	1.241152	**2.135932**	**2.248385**	**2.379845**	**2.933165**
		IT2 Fuzzy	**2.915063**	1.241152	1.689376	**2.248385**	**2.260518**	**2.986203**
	Max Dependency	Classical	**2.477756**	**4.720209**	**2.335226**	**2.593460**	**5.244840**	**4.978336**
		Neighborhood	**1.940231**	**3.647312**	1.628484	**2.573813**	1.475539	**3.857665**
		T1 Fuzzy	**2.754230**	**3.276650**	1.689376	**2.215969**	**1.907565**	**3.271380**
		IT2 Fuzzy	**2.070948**	**2.190386**	1.689376	**2.822734**	**2.160387**	**3.271380**
	MRMS	Classical	**5.897814**	**4.720209**	1.496205	**1.947889**	**3.878816**	1.732453
		Neighborhood	**3.627165**	**4.016121**	1.351572	**1.839908**	1.480069	1.250769
		FR-MRMS	1.565908	0.747411	0.723614	1.302457	1.139661	0.877816
		IT2FR-MRMS	1.245149	0.953282	0.651127	0.861573	1.139661	0.625755
	FR-SFSFEx		(79.47/9, 6.74)	(76.26/13, 7.54)	(85.12/12, 6.36)	(85.37/12, 8.94)	(85.37/16, 6.88)	(91.63/12, 5.73)
SVM	Max Relevance	Classical	**6.812529**	**3.813885**	**2.522248**	1.700252	**5.299347**	**4.423587**
		Neighborhood	**4.183872**	**3.200821**	1.793523	**1.882382**	**2.521177**	**4.189523**
		T1 Fuzzy	1.561457	**2.113385**	1.505271	**2.001256**	**2.115388**	**3.771727**
		IT2 Fuzzy	**2.404573**	**2.669677**	1.505271	**2.001256**	**2.147030**	**2.587989**
	Max Dependency	Classical	**3.665454**	**4.190779**	1.420710	**1.965739**	**5.462011**	**5.361321**
		Neighborhood	**3.082310**	**3.412096**	1.351672	**2.568160**	**2.073929**	**3.746171**
		T1 Fuzzy	**3.782355**	**3.360922**	1.069037	**2.449639**	**2.107118**	**2.819480**
		IT2 Fuzzy	**3.816683**	**3.360922**	0.735736	**2.860232**	**2.019228**	**2.819480**
	MRMS	Classical	**5.634003**	**3.692035**	1.054487	**1.965739**	**3.551048**	**2.685945**
		Neighborhood	**2.785658**	**2.960939**	0.740928	**1.927843**	**2.169097**	**2.233635**
		FR-MRMS	1.360146	**2.015074**	0.111708	1.440493	1.178625	1.618333
		IT2FR-MRMS	1.382839	**2.024161**	0.000000	1.077223	1.055017	1.618333
	FR-SFSFEx		(80.38/10, 6.88)	(72.65/13, 7.84)	(83.23/10, 8.33)	(78.55/13, 8.76)	(85.87/8, 8.56)	(94.72/9, 3.88)
C4.5	Max Relevance	Classical	**2.152366**	**2.008951**	**3.315868**	**2.639004**	**4.169341**	**5.257245**
		Neighborhood	**2.464462**	**2.415512**	**3.217395**	**2.498178**	**3.650701**	**3.434978**
		T1 Fuzzy	**2.684401**	**1.852976**	**2.332207**	**2.347948**	**3.970696**	**3.443705**
		IT2 Fuzzy	**2.847140**	**1.852976**	**1.869905**	**2.287002**	**3.970696**	**2.058608**
	Max Dependency	Classical	**3.013972**	**3.450607**	**3.525669**	**2.635894**	**4.169341**	**4.411366**
		Neighborhood	**2.78730**	**2.589355**	**2.785982**	**2.805541**	**3.650701**	**1.866983**
		T1 Fuzzy	**2.152366**	**1.827592**	**2.638829**	**1.989263**	**3.970696**	**3.546786**
		IT2 Fuzzy	**2.023615**	1.663538	**2.842536**	**4.833675**	**3.094047**	**2.638147**
	MRMS	Classical	**2.684401**	**2.366537**	**2.324960**	**2.722161**	**4.103795**	1.583788
		Neighborhood	1.441457	1.307066	**2.105411**	1.577296	**2.193954**	1.300467
		FR-MRMS	1.092949	0.778080	1.255682	1.591290	1.292966	**2.082604**
		IT2FR-MRMS	1.203051	0.837262	0.992084	1.056489	0.931204	1.650098
	FR-SFSFEx		(78.94/14, 9.48)	(74.38/10, 10.22)	(87.47/9, 6.88)	(85.66/8, 8.96)	(80.76/13, 5.87)	(91.76/12, 6.38)

Satimage, Segmentation, Leukemia II, and Breast II data based on training-testing and Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features data based on 10-fold CV. For 10-fold CV, the best average accuracy of 10 folds is presented. The results and subsequent discussions are analyzed in this table with respect to classification accuracy of K-NN, SVM, and C4.5. The best test accuracy obtained from all possible parameters values on each data set is compared with the test accuracy corresponding to best training accuracy and that for optimum parameters.

All the results reported in Table 5.2 confirm that the test accuracy obtained using optimum parameters is higher than the test accuracy corresponding to best training accuracy and comparable with the best test accuracy in most of the cases, irrespective of the rough sets, classifiers, and data sets used. Out of 30 cases each, the test accuracy obtained using the proposed technique is exactly same with the best test accuracy in 21, 20, and 25 cases for classical, neighborhood, and fuzzy-rough sets, respectively. For fuzzy-rough sets, the test accuracy corresponding to best training is better than that of the proposed technique in only 1 case, while for classical and neighborhood rough sets, it is only 3 and 4 cases, respectively.

Similarly, Table 5.3 presents the performance of class separability index, DB index, Dunn index, and FFE index on Satimage, Segmentation, Leukemia II, and Breast II data based on training-testing and Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features data based on 10-fold CV. All the results reported in Table 5.3 confirm that the test accuracy obtained using optimum parameters is higher than the test accuracy corresponding to best training accuracy and comparable with the best test accuracy in most of the cases, irrespective of the indices, classifiers, and data sets used. Out of 30 cases each, the test accuracy obtained using the proposed technique is exactly same with the best test accuracy in 19, 20, 20, and 20 cases for class separability index, DB index, Dunn index, and FFE index, respectively. The test accuracy corresponding to best training is better than that of the proposed technique in only 3, 2, 1, and 2 cases, for class separability index, DB index, Dunn index, and FFE index, respectively.

5.3.3 Performance of Various Rough Set Based SFSFEx

In case of dimensionality reduction, the reduced feature set is always relative to a certain feature evaluation index. In general, different evaluation indices may lead to different reduced feature subsets. To establish the effectiveness of fuzzy-rough sets over Pawlak's or classical and neighborhood rough sets, extensive experiments are done on various data sets. Tables 5.4, 5.5, and 5.6 present the comparative performance of different rough set models for simultaneous feature selection and extraction task. The results and subsequent discussions are presented in this table with respect to the classification accuracy of the K-NN, SVM, and C4.5 on test samples considering the optimum parameter values.

Table 5.10: Statistical Significance Test of 10-fold CV with Different Indices on Colon, Breast I, Lung, Leukemia I, Isolet, and MFeat

Different Classifiers	Different Criteria	Different Indices	Different Data Sets					
			Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	Max Relevance	Class Separability	**2.383157**	**5.001845**	**2.919045**	**2.774585**	**6.928629**	**4.954369**
		DB Index	1.668868	**3.933162**	**2.776771**	**2.340863**	**2.712199**	**5.554501**
		Dunn Index	**3.228138**	**3.524622**	**2.587852**	**3.098229**	**2.880131**	**5.242585**
		FFE Index	**2.376669**	**3.933162**	**3.220075**	**3.121797**	**5.471720**	**5.352032**
	Max Dependency	Class Separability	**2.865581**	**5.283480**	**3.222570**	**3.130374**	**7.229573**	**5.729549**
		DB Index	**3.047862**	**3.743321**	**3.222570**	**2.629033**	**3.779499**	**5.242585**
		Dunn Index	**3.047862**	**4.388635**	**2.521078**	**2.593460**	**3.550241**	**5.896730**
		FFE Index	**3.532858**	**3.960200**	**3.329620**	**2.848638**	**5.261860**	**5.601838**
	MRMS	Class Separability	1.684669	**4.635743**	**2.623857**	**1.947889**	**6.745449**	**3.320599**
		DB Index	1.456453	**3.647312**	**3.125884**	1.739324	**3.113252**	**2.494107**
		Dunn Index	1.456453	**3.201125**	**1.905957**	1.534782	**3.394975**	**2.968613**
		FFE Index	**2.026509**	**3.629771**	**2.840504**	**2.573813**	**5.150028**	**3.455508**
	FR-SFSFEx		(79.47/9, 6.74)	(76.26/13, 7.54)	(85.12/12, 6.36)	(85.37/12, 8.94)	(85.37/16, 6.88)	(91.63/12, 5.73)
SVM	Max Relevance	Class Separability	**2.265639**	**3.306459**	**2.053740**	**2.801783**	**7.200995**	**4.378930**
		DB Index	1.802228	**3.200821**	1.735797	**2.008702**	**2.785033**	**4.621799**
		Dunn Index	1.802228	**2.345625**	1.693846	**2.549775**	**3.137590**	**4.100271**
		FFE Index	**2.184320**	**3.306459**	**2.281644**	**2.749614**	**3.228860**	**5.437592**
	Max Dependency	Class Separability	**2.351697**	**3.465187**	**2.179994**	**2.630282**	**7.007223**	**5.588556**
		DB Index	**2.265639**	**3.466301**	1.660871	**2.581177**	**4.030787**	**5.563348**
		Dunn Index	**2.184320**	**3.306459**	1.455513	**2.240064**	**4.259593**	**4.671847**
		FFE Index	**2.659827**	**4.605708**	**2.179994**	**2.983488**	**3.934268**	**5.019070**
	MRMS	Class Separability	**1.992707**	**4.467398**	**1.982659**	**2.240064**	**6.375923**	**4.834181**
		DB Index	1.636817	**3.143343**	1.809676	**2.392234**	**3.332709**	**2.615068**
		Dunn Index	1.636817	**4.190779**	1.503171	**1.938011**	**3.620115**	**2.951243**
		FFE Index	1.723036	**3.258115**	**1.883814**	**2.359174**	**4.189343**	**2.795034**
	FR-SFSFEx		(80.38/10, 6.88)	(72.65/13, 7.84)	(83.23/10, 8.33)	(78.55/13, 8.76)	(85.87/8, 8.56)	(94.72/9, 3.88)
C4.5	Max Relevance	Class Separability	**1.884688**	**2.686117**	**3.716593**	**2.999429**	**8.074341**	**5.001151**
		DB Index	1.579093	**2.610751**	**3.495789**	**2.697040**	**4.833382**	**4.729479**
		Dunn Index	1.640377	**2.220080**	**3.207414**	**2.429224**	**6.356946**	**4.107129**
		FFE Index	**2.109368**	**3.210648**	**3.960769**	**2.935767**	**8.191139**	**4.521711**
	Max Dependency	Class Separability	**2.378827**	**2.917093**	**3.551877**	**2.914244**	**7.694716**	**4.397692**
		DB Index	**2.539836**	**3.130904**	**2.969874**	**3.050619**	**4.082432**	**4.048513**
		Dunn Index	**2.340718**	**3.073166**	**2.791939**	**2.867239**	**6.180868**	**4.361631**
		FFE Index	**2.558310**	**3.570586**	**3.727293**	**3.177182**	**6.481413**	**4.998472**
	MRMS	Class Separability	1.313486	**2.730731**	**2.912526**	**2.622161**	**6.442835**	**2.379072**
		DB Index	1.136933	**1.838866**	**3.228388**	**1.989263**	**4.853413**	1.513039
		Dunn Index	1.275196	**1.936887**	**2.643263**	**2.171076**	**6.930507**	1.669370
		FFE Index	1.313486	**2.366537**	**3.187757**	**2.551752**	**7.077409**	1.277499
	FR-SFSFEx		(78.94/14, 9.48)	(74.38/10, 10.22)	(87.47/9, 6.88)	(85.66/8, 8.96)	(80.76/13, 5.87)	(91.76/12, 6.38)

Table 5.11: Statistical Significance Test of 10-fold CV with Different Methods on Colon, Breast I, Lung, Leuke I, Isolet, and MFeat

Different Classifiers	Different Methods / Algorithms	Different Data Sets					
		Colon	Breast I	Lung	Leukemia I	Isolet	Multiple Features
K-NN	InfoGain	**1.927041**	**2.211637**	**4.184433**	**2.007099**	**2.206781**	**6.192207**
	Classical mRMR	**2.101269**	**2.674028**	**4.184433**	**2.407230**	**2.307431**	**3.420496**
	Fuzzy-Rough mRMR	1.190758	**2.913112**	0.723614	**3.833657**	1.094707	**2.128354**
	RELIEF	**2.635781**	**4.720209**	**3.374015**	**2.237650**	**2.254982**	**3.062063**
	SIMBA	**2.569383**	**3.895184**	**2.389338**	**1.948982**	**2.622353**	**2.743902**
	Wrapper	**4.691615**	**3.765217**	**4.792462**	**5.064917**	**5.878739**	**5.939340**
	WFS	**5.665529**	**3.815696**	**4.077563**	**5.508722**	**7.940473**	**5.436725**
	CFS	**4.708075**	**3.941512**	**3.179516**	**4.126672**	**5.899789**	**6.691480**
	PCA	**2.009646**	**2.485125**	1.485332	1.701177	**3.316603**	**2.566203**
	ICA	**2.057557**	**2.289488**	1.348511	**2.042703**	**2.382497**	**2.089823**
	LDA	**2.781286**	**2.731795**	1.515252	1.543931	**2.548324**	**2.034916**
	PLS-DA	**2.142795**	**2.595338**	1.189446	1.316424	0.728391	0.722590
	FR-SFSFEx	(79.47/9, 6.74)	(76.26/13, 7.54)	(85.12/12, 6.36)	(85.37/12, 8.94)	(85.37/16, 6.88)	(91.63/12, 5.73)
SVM	InfoGain	**2.087280**	1.363115	1.438786	**1.981385**	**1.827596**	**4.365410**
	Classical mRMR	**1.899951**	1.355344	**1.829146**	**2.085964**	**2.139732**	**3.659920**
	Fuzzy-Rough mRMR	1.120983	1.336913	1.551083	1.446623	1.045784	**2.110213**
	RELIEF	**3.386038**	**3.820543**	**2.219428**	1.787536	1.785749	**3.100766**
	SIMBA	**3.210315**	**3.651465**	**2.073194**	**2.081407**	1.807090	**2.931042**
	Wrapper	**4.098321**	1.327914	**5.549066**	**3.177238**	**4.146642**	**8.244624**
	WFS	**4.292632**	0.058022	**5.649645**	**3.601210**	**6.084982**	**5.136230**
	CFS	**7.719735**	**2.025053**	**2.603340**	**3.238236**	**4.855792**	**10.675251**
	PCA	**3.031863**	**2.172915**	1.650320	1.499725	1.314733	**3.191908**
	ICA	**3.176619**	**2.333945**	**1.979773**	1.410990	0.834486	**2.814180**
	LDA	**3.364997**	**2.746246**	1.498082	1.301213	1.138184	**3.123739**
	PLS-DA	1.639628	**2.157630**	1.018730	**2.031705**	0.462608	1.025483
	FR-SFSFEx	(80.38/10, 6.88)	(72.65/13, 7.84)	(83.23/10, 8.33)	(78.55/13, 8.76)	(85.87/8, 8.56)	(94.72/9, 3.88)
C4.5	InfoGain	**2.684401**	**3.038348**	**4.533872**	**2.622161**	**2.652064**	**3.728899**
	Classical mRMR	1.399136	**3.189041**	**2.808826**	**2.101806**	**2.437880**	**2.949020**
	Fuzzy-Rough mRMR	0.981068	**2.819087**	**2.545325**	1.807761	**2.429225**	**3.060925**
	RELIEF	**2.202249**	**2.807203**	**3.710464**	**1.961276**	**3.377103**	**3.082107**
	SIMBA	**1.854923**	**2.780624**	**3.270627**	**1.989263**	**2.924033**	**3.140578**
	Wrapper	**3.774255**	**2.044836**	**3.071884**	**3.298657**	**5.999121**	**3.279610**
	WFS	**2.266232**	**2.138158**	**2.629354**	**5.320088**	**9.993188**	**2.486326**
	CFS	**3.888826**	**3.989491**	**5.424825**	**5.377881**	**6.792364**	**4.863637**
	PCA	**1.884688**	1.380031	**2.575765**	1.577296	**4.345458**	1.600749
	ICA	**2.044073**	1.403294	**2.994569**	1.746370	0.000000	1.116877
	LDA	1.488738	**2.101309**	**2.626482**	**2.096552**	-0.556106	1.092931
	PLS-DA	1.251166	1.381003	**2.206913**	**2.32812**	-0.202251	1.544502
	FR-SFSFEx	(78.94/14, 9.48)	(74.38/10, 10.22)	(87.47/9, 6.88)	(85.66/8, 8.96)	(80.76/13, 5.87)	(91.76/12, 6.38)

All the training-testing, 10-fold CV, and bootstrap $\mathcal{E}_{\text{BS}}^{.632}$ are performed to compute the classification accuracy of the K-NN, SVM, and C4.5. Tests of significance are performed for the inequality of means (of best classification accuracy of the K-NN, SVM, and C4.5) obtained using the fuzzy-rough sets and other rough sets. Since both mean pairs and the variance pairs are unknown and different, a generalized version of t-test is used here, which is already described in Section 3.4.1 of Chapter 3. Tables 5.4, 5.5, and 5.6 report the individual means and standard deviations, and the value of test statistic computed. In case of 10-fold CV, the best means and corresponding standard deviations of classification accuracy obtained in different folds are computed for Colon, Breast I, Lung, Leukemia I, Isolet, and Multiple Features data sets. The corresponding tabled value is 1.81 at an error probability level of 0.05. If the computed value is greater than the tabled value, the means are significantly different, which are marked bold in the Tables 5.4, 5.5, and 5.6.

From the results reported in Tables 5.4, 5.5, and 5.6, it can be seen that the proposed FR-SFSFEx method attains maximum classification accuracy of the K-NN, SVM, and C4.5 in most of the cases. Out of 12 cases of training-testing, the proposed FR-SFSFEx method achieves highest classification accuracy in 10 cases, while that with CR-SFSFEx (classical rough sets based SFSFEx) attains it only in 2 cases. On the other hand, among the 36 comparisons of 10-fold CV, the proposed FR-SFSFEx method provides significantly better results in 21 cases and better results but not significantly in 11 cases, while significantly better results are achieved only in 3 cases using NR-SFSFEx (neighborhood rough sets based SFSFEx). All other cases, the performance of different rough sets is same. Experiments are also done on the data sets using bootstrap $\mathcal{E}_{\text{BS}}^{.632}$ method, where the results are represented in the same table. Here, among the 18 comparisons, the proposed FR-SFSFEx method provides significantly better results in each and every cases. In brief, out of total 48 cases, the CR-SFSFEx and NR-SFSFEx attain higher classification accuracy than the FR-SFSFEx in 2 and 3 cases, respectively. All other cases, FR-SFSFEx provide higher or comparable classification accuracy, irrespective of the data sets, experimental setup, and classifiers used. The proposed FR-SFSFEx is also compared with various rough sets based feature selection methods which are reported in Table 3.1 of Chapter 3, Tables 5.4, 5.5, and 5.6 for training-testing samples. From Table 3.1 of Chapter 3, Tables 5.4, 5.5, and 5.6, it can be seen that, out of 12 cases of training-testing, the proposed FR-SFSFEx method achieves highest classification accuracy in 10 cases, while IT2FR-MRMS attains it only in 2 cases. The performance of the FR-SFSFEx method is also compared with various rough sets based feature selection methods for 10-fold CV and bootstrap $\mathcal{E}_{\text{BS}}^{.632}$ method. Table 5.9 reflects the value of test statistic computed corresponding to 10-fold CV, where the individual means, number of features selected, and standard deviations are reported for FR-SFSFEx. From Table 5.9, it can be seen that, out of 216 cases of 10-fold CV, the proposed FR-SFSFEx method provides significantly better results in 154 cases, and better

results but not significantly in other 62 cases. Table 4.7 of Chapter 4 and Table 5.8 present the bootstrap $\mathcal{E}_{\mathrm{BS}}^{.632}$ results, where, out of 18 cases, the proposed FR-SFSFEx provides best results in 16 cases. The proposed method also provides higher classification accuracy than the Max-Relevance, Max-Dependency, and MRMS criteria in all cases, irrespective of the classifiers, rough sets, and data sets used. The better performance of the FR-SFSFEx is achieved due to the fact that it can capture uncertainties associated with the data more accurately.

5.3.4 Performance of Various Non-Rough Set Based SFSFEx

All the training-testing, 10-fold CV, and bootstrap $\mathcal{E}_{\mathrm{BS}}^{.632}$ are performed to compare the classification accuracy of FR-SFSFEx with non-rough set based SFSFEx, namely, ClsSep-SFSFEx (Class Separability index based SFSFEx), DB-SFSFEx (DB index based SFSFEx), Dunn-SFSFEx (Dunn index based SFSFEx), and FFEI-SFSFEx (FFE index based SFSFEx) with respect to K-NN, SVM, and C4.5. From the results reported in Tables 5.4, 5.5, and 5.6, it can be seen that the proposed FR-SFSFEx method attains maximum classification accuracy of the K-NN, SVM, and C4.5 in most of the cases. Out of 12 cases of training-testing, the proposed FR-SFSFEx method achieves highest classification accuracy in all the cases. On the other hand, among the 72 comparisons of 10-fold CV, the proposed FR-SFSFEx method provides significantly better results in 52 cases and better results but not significantly in 17 cases, while ClsSep-SFSFEx and DB-SFSFEx achieve better result, but not significantly in only 1 and 2 cases, respectively. In all other cases, the performance of different indices is same. Experiments are also conducted on the data sets using bootstrap $\mathcal{E}_{\mathrm{BS}}^{.632}$ method, where the results are represented in the same table. Here, among the 18 comparisons, the proposed FR-SFSFEx method provides significantly better results in 17 cases, while Dunn-SFSFEx achieves it in only one case. In brief, out of total 48 cases, the ClsSep-SFSFEx, DB-SFSFEx, and Dunn-SFSFEx attain higher classification accuracy than the FR-SFSFEx in only 1, 2, and 1 cases, respectively. In all other cases, FR-SFSFEx provide higher or comparable classification accuracy, irrespective of the data sets, experimental setup, and classifiers used. The proposed FR-SFSFEx is also compared with various non-rough set based feature selection methods which are reported in Table 3.1 of Chapter 3, Tables 5.4, 5.5, and 5.6 for training-testing data sets. From Table 3.1 of Chapter 3, Tables 5.4, 5.5, and 5.6, it can be seen that the proposed FR-SFSFEx method achieves highest classification accuracy in all the cases of training-testing. The performance of the FR-SFSFEx method is also compared with various non-rough sets based feature selection methods for 10-fold CV and bootstrap $\mathcal{E}_{\mathrm{BS}}^{.632}$ method. Table 5.10 reflects the experimental results corresponding to 10-fold CV, where the individual means, number of features selected, and standard deviations are reported for FR-SFSFEx. From Table 5.10, it can be seen that, out of 216 cases of 10-fold CV, the proposed FR-SFSFEx method

provides significantly better results in 190 cases, and better results but not significantly in other 26 cases. Table 4.8 of Chapter 4 and Table 5.8 present the bootstrap $\mathcal{E}_{\mathrm{BS}}^{632}$ results, where the proposed FR-SFSFEx provides best results in all of the cases.

5.3.5 Performance of Different Algorithms

Table 3.1 of Chapter 3, Tables 5.7, 5.8, and 5.11 compare the performance of the proposed FR-SFSFEx algorithm with that of different existing feature selection and extraction algorithms on various data sets. From the training-testing results reported in Table 3.1 of Chapter 3 and Table 5.7, it is seen that the proposed dimensionality reduction method achieves highest classification accuracy of K-NN, SVM, and C4.5 in 11 cases, out of total 12 cases, while the PCA attains highest classification accuracy in only 1 case. The performance of the FR-SFSFEx method is also compared with various other feature selection and feature extraction algorithms for 10-fold CV and bootstrap $\mathcal{E}_{\mathrm{BS}}^{632}$ method. From the results reported in Table 5.11, it can be seen that, out of 144 cases of 10-fold CV, the proposed FR-SFSFEx method provides significantly better results in 104 cases, and better results but not significantly in 38 cases. Table 4.9 of Chapter 4 and Table 5.8 present the bootstrap $\mathcal{E}_{\mathrm{BS}}^{632}$ results, where the proposed FR-SFSFEx provides best results results in 17 cases, out of 18 cases. Also, the proposed method can potentially yield significantly better results than the existing algorithms.

5.3.6 Importance of Proposed Feature Extraction Method

To extract a new feature $\overline{\mathcal{A}}_i$ from the dispensable set D_i corresponding to the feature $\mathcal{A}_i$, that is, step 4(b) of the proposed FR-SFSFEx algorithm, one may consider the conventional feature extraction techniques like PCA, ICA, and PLS-DA instead of the feature extraction method presented in Section 5.2.1. In order to establish the importance of proposed feature extraction technique over the conventional PCA, ICA, and PLS-DA in step 4(b), extensive experimentation is done on the data sets mentioned earlier. In Tables 5.12, 5.13, and 5.14, PCA-SFSFEx, ICA-SFSFEx, and PLS-DA-SFSFEx represent, respectively, the algorithms generated if 4(b) of FR-SFSFEx algorithm is replaced by PCA, ICA, and PLS-DA. From the results reported in Table 5.12, it can be seen that, out of total 12 cases, the proposed technique provides highest classification accuracy in terms of K-NN, SVM, and C4.5 in 11 cases. However, the PLS-DA-SFSFEx attains highest classification accuracy in 1 case.

Similarly, all the results reported in Tables 5.13 and 5.14 confirm that the proposed FR-SFSFEx method performs significantly better than that of other algorithms in 46 cases out of total 54 cases, while the PLS-DA-SFSFEx achieves significantly higher accuracy in only 1 case. In all other cases, the performance of different algorithms is comparable. The better performance of the proposed FR-SFSFEx technique is achieved due to the fact

Table 5.12: Performance Analysis of Different SFSFEx Methods on Satimage, Segmentation, Leukemia II, and Breast II

Different Data Sets	K-NN				SVM				C4.5			
	PCA-SFSFEx	ICA-SFSFEx	PLS-DA-SFSFEx	FR-SFSFEx	PCA-SFSFEx	ICA-SFSFEx	PLS-DA-SFSFEx	FR-SFSFEx	PCA-SFSFEx	ICA-SFSFEx	PLS-DA-SFSFEx	FR-SFSFEx
Satimage	83.03	82.89	85.60	**87.55**	82.75	82.68	85.27	**87.35**	82.73	82.74	85.10	**87.20**
Segmentation	81.80	81.64	84.39	**86.24**	87.53	87.30	90.40	**92.33**	85.74	85.60	88.51	**90.33**
Leukemia II	89.59	89.71	**94.72**	94.64	88.88	89.06	91.77	**93.75**	88.83	88.65	91.65	**93.75**
Breast II	89.74	89.79	92.56	**94.73**	89.93	89.63	92.46	**94.73**	94.97	94.99	97.89	**100.0**

Table 5.13: Analysis of SFSFEx based on PCA, ICA, PLS-DA and Proposed SFSFEx Method Using 10 Fold CV

Different Methods	Different Statistics	Colon			Breast I			Lung		
		K-NN	SVM	C4.5	K-NN	SVM	C4.5	K-NN	SVM	C4.5
FR-SFSFEx	Mean	79.47	80.38	78.94	76.26	72.65	74.38	85.12	83.23	87.47
	Std Dev	6.74	6.88	9.48	7.54	7.84	10.22	6.36	8.33	6.88
PCA-SFSFEx	Mean	71.04	70.04	71.14	67.48	69.56	68.73	71.15	69.73	70.62
	Std Dev	6.08	5.47	5.46	6.55	8.83	8.15	7.92	8.45	7.78
	Computed	**2.93**	**3.72**	**2.25**	**2.78**	0.83	1.37	**4.35**	**3.60**	**5.13**
ICA-SFSFEx	Mean	71.08	67.07	68.53	70.45	68.63	68.26	71.19	70.33	67.29
	Std Dev	6.86	7.53	7.60	5.10	5.12	6.86	7.83	8.55	5.03
	Computed	**2.76**	**4.13**	**2.71**	**2.02**	1.36	1.57	**4.37**	**3.42**	**7.49**
PLS-DA-SFSFEx	Mean	70.30	68.90	72.07	70.25	69.44	70.47	70.54	71.41	71.47
	Std Dev	7.33	6.18	7.79	7.13	8.13	6.09	8.10	5.90	6.73
	Computed	**2.91**	**3.93**	1.77	**1.83**	0.90	1.04	**4.48**	**3.66**	**5.26**

Table 5.14: Analysis of SFSFEx based on PCA, ICA, PLS-DA and Proposed SFSFEx Method Using 10 Fold CV

Different Methods	Different Statistics	Leukemia I			Isolet			Multiple Features		
		K-NN	SVM	C4.5	K-NN	SVM	C4.5	K-NN	SVM	C4.5
FR-SFSFEx	Mean	85.37	78.55	85.66	85.37	85.87	80.76	91.63	94.72	91.76
	Std Dev	8.94	8.76	8.96	6.88	8.56	5.87	5.73	3.88	6.38
PCA-SFSFEx	Mean	72.33	70.55	71.05	63.32	62.79	58.85	67.63	67.89	70.03
	Std Dev	5.73	7.91	5.57	7.43	5.11	5.76	8.51	5.91	7.34
	Computed	**3.88**	**2.15**	**4.38**	**6.88**	**7.32**	**8.42**	**7.40**	**12.00**	**7.07**
ICA-SFSFEx	Mean	72.73	69.43	73.04	61.09	62.71	60.06	67.84	69.09	71.85
	Std Dev	7.10	6.59	6.47	7.12	8.42	6.40	8.57	8.81	5.64
	Computed	**3.50**	**2.63**	**3.61**	**7.75**	**6.10**	**7.54**	**7.29**	**8.42**	**7.40**
PLS-DA-SFSFEx	Mean	71.22	73.77	71.88	62.58	61.72	63.81	69.04	69.38	72.02
	Std Dev	7.75	8.71	8.64	7.13	5.16	6.65	6.23	8.56	5.83
	Computed	**3.78**	1.22	**3.50**	**7.27**	**7.64**	**6.04**	**8.43**	**8.53**	**7.22**

Table 5.15: Execution Time (in second) of Different Methods for Various Data Sets

Different Indices	Different Benchmark Data Sets									
	Satimage	Segmen	Leuke II	Breast II	Colon	Breast I	Lung	Leuke I	Isolet	Mult Feat
Wrapper	2.0E+02	1.0E+02	5.5E+02	7.8E+02	2.3E+02	9.8E+02	5.9E+02	5.8E+02	4.8E+02	3.9E+02
WFS	8.4E+01	1.1E+02	1.3E+02	5.5E+02	4.1E+02	1.2E+03	6.5E+02	8.7E+02	3.0E+02	4.1E+02
CFS	5.0E+01	1.7E+02	2.3E+01	5.1E+02	2.6E+02	1.3E+03	7.5E+02	1.3E+03	1.4E+02	1.2E+02
PLS-DA	9.8E+03	1.2E+03	6.2E+03	7.2E+04	4.2E+04	2.4E+04	7.7E+04	5.0E+04	1.9E+04	4.4E+04
FR-SFSFEx	2.8E+01	3.1E+01	6.3E+01	7.3E+01	6.4E+00	4.3E+01	4.5E+01	8.3E+01	4.9E+01	1.0E+01

that the PCA, ICA, and PLS-DA do not take into account the relevance of the features, rather, they only consider the variances of different features present in the dispensable set, to generate a new feature. In effect, they may fail to extract relevant features.

5.3.7 Execution Time

Table 3.8 of Chapter 3 and Table 5.15 report the execution time of different algorithms. The significantly lesser time of the proposed algorithm is achieved due to its low computational complexity. The better performance of the proposed method is achieved due to the fact that it provides an efficient way to simultaneously select and extract features for classification. In effect, a reduced set of features having maximum relevance and significance is being obtained using the proposed method.

5.4 Conclusion

Dimensionality reduction using feature selection and/or feature extraction is one of the important problems in pattern recognition and data mining, keeping in mind the explosive growth of available information. In this regard, this chapter presents a novel dimensionality reduction method, integrating judiciously the theory of fuzzy-rough sets and merits of both attribute selection and feature extraction. An efficient algorithm is introduced by performing simultaneous feature selection and extraction. It uses the concept of fuzzy-rough feature significance for finding significant and relevant features from real valued data sets. Finally, the effectiveness of the proposed method is presented, along with a comparison with other related algorithms, on several real life data sets. This formulation is geared towards maximizing the utility of fuzzy-rough sets, feature selection, and feature extraction with respect to knowledge discovery tasks. Through these investigations and experiments, the potential utility of fuzzy-rough sets for dimensionality reduction is demonstrated.

Next chapter addresses the problem of clustering, which is another important problem of pattern recognition, machine learning, and data mining. A new clustering algorithm, termed as rough hypercuboid based interval type-2 fuzzy c-means, is presented integrating

judiciously the theory of rough hypercuboid approach, interval type-2 fuzzy sets, and c-means algorithm, to approximately manage the uncertainty present in a data set.

Chapter 6

Rough Hypercuboid Based IT2 Fuzzy C-Means Algorithm

6.1 Introduction

Clustering has retained high interest in a wide range of engineering and scientific disciplines like pattern recognition, data mining, machine learning, bioinformatics, web intelligence, remote sensing, communications, and computer vision, where objects are grouped into a number of clusters according to their similarities in multidimensional spaces [112]. The grouping is done in such a way that the degree of association between objects from the same cluster is maximum and minimum for the objects from different clusters. The classical k-means or hard c-means (HCM) algorithm [112] is one of the primitive and most widely used objective function based clustering algorithms, which minimizes the sum of the distances between the samples and the corresponding centroid. Each sample has membership in only one of the clusters.

Uncertainty is one of the major problems in real life data analysis. Some of the sources of this uncertainty include incompleteness and vagueness in cluster definition. Fuzzy set theory is an extension of the classical set theory, which permits gradual membership values for elements in a set, relaxing the requirement of the HCM. In this background, fuzzy c-means (FCM) algorithm, originally proposed by Dunn [113] and then modified by Bezdek [48], is a widely used and efficient partition-based clustering technique, where the memberships of the objects decrease as their distance from the cluster center increases. When the clusters are not precisely defined, and having overlapping boundaries, the FCM is a better choice for clustering compared to the HCM. It provides more accurate results for the data that belong to the intersection of multiple clusters. The FCM uses probabilistic approach for clustering. However, for real life data sets, where noise and outliers are unavoidable, the FCM forces those undesired objects to put in one or more clusters. To

resolve this issue, Krishnapuram and Keller proposed possibilistic c-means (PCM) [261], which generates the clusters by calculating the distance between any object to that cluster, and does not consider the distance with other clusters. The PCM performs well in noisy environment or data set with outliers; but the algorithm is very sensitive to initial prototypes, and often produces coincident clusters. Pal et al. [383] proposed a clustering algorithm, which combines both fuzzy (probabilistic) and possibilistic memberships.

The theory of rough sets, proposed by Pawlak, is a very useful paradigm, which can be used to deal with uncertainty, incompleteness, and vagueness associated with data [414]; and hence has been widely used for feature selection [414] and clustering in approximation spaces [320, 321, 330]. Combination of fuzzy sets and rough sets provides an effective way to solve this uncertainty problems [110, 220, 571]. The generalized theories of rough-fuzzy computing have been applied successfully to feature selection of real valued data [199, 220, 223, 314, 323, 500]. Integrating both rough and fuzzy sets, Maji and Pal proposed rough-fuzzy c-means (RFCM) algorithm [321], encapsulating two related and complementary, but distinct concepts used to represent the uncertainty in knowledge; vagueness (used in fuzzy set), and indiscernibility (used in rough sets). Each cluster in the RFCM consists of two disjoint regions, a crisp lower approximation, surrounded by an uncertain fuzzy boundary. Each object in lower approximation solely belongs to that cluster and has same weight, resulting a similar influence on the corresponding cluster and centroid updation. The algorithm provides the boundary region with gradual membership values, which can group the objects well in a data set with the presence of uncertainty. A cluster can be represented by its center, lower approximation and boundary region. A generalized hybrid algorithm, termed as rough-fuzzy possibilistic c-means (RFPCM), has also been proposed in [321]. The algorithm has been further extended to robust rough-fuzzy c-means (rRFCM) [330], where possibilistic lower approximation is surrounded by a probabilistic boundary region for any cluster. The possibilistic lower approximation helps to discover clusters having various shapes.

However, the FCM, RFCM, RFPCM or rRFCM algorithms use type-1 (T1) fuzzy sets. Although the researchers found some limitations of T1 fuzzy sets, most of the contributions in this domain are still restricted to this theory. As discussed in Chapter 4, Mendel [342] pointed out several sources of uncertainties present in T1 fuzzy logic systems. All these uncertainties ultimately converge into uncertainties about the membership functions of fuzzy sets. The T1 fuzzy set has been generalized to type-2 (T2) fuzzy set [519] by Zadeh [555]. It uses membership grades that are also fuzzy. The T2 fuzzy set is useful for the scenario where the exact membership function for the fuzzy set is costly to determine. Three-dimensional membership function of the T2 fuzzy set provides more degrees of freedom than that of the T1 fuzzy set, which uses two-dimensional membership function [347]. Rhee and Hwang proposed T2 fuzzy c-means algorithm (T2FCM) [437] by assigning

membership grades to the T1 memberships. Liang and Mendel [283] and Mendel et al. [345] have shown that interval T2 (IT2) fuzzy set is more useful in practical applications, due to its lower computational complexity than the T2 fuzzy set. In this background, an IT2 fuzzy c-means (IT2FCM) algorithm has been proposed in [206].

The IT2 fuzzy set is also successfully applied in various pattern recognition applications [316, 531], combining it with rough set theory. Hence, the combination of IT2 fuzzy set and rough sets can be a promising technique for clustering, if the integration is done in a cooperative, rather than a competitive, manner [316]. The integration is expected to enable the clustering process artificially more intelligent. Incorporation of IT2 fuzzy set will help to handle efficiently overlapping clusters where the determination of exact membership function is difficult, while the concept of lower and upper approximations of rough sets will deal with uncertainty, vagueness, and incompleteness in cluster definitions. Since the integrated approach has the capability of providing a stronger paradigm for uncertainty handling [316, 531], it has greater promise in clustering, where IT2 fuzzy set [206, 437] and/or rough sets [321, 330] are being effectively used and proved to be successful.

In this regard, this chapter introduces a hybrid clustering algorithm [145], termed as rough hypercuboid based interval type-2 fuzzy c-means (RH-IT2FCM). It integrates judiciously the merits of rough hypercuboid approach, probabilistic membership of IT2 fuzzy sets, and c-means algorithm. While the concept of lower approximation and boundary region of rough sets deals with uncertainty, incompleteness, and vagueness in cluster definition, the use of fuzzy membership of IT2 fuzzy sets in the boundary region enables efficient handling of overlapping partitions in uncertain environment. One of the important issues of rough set based clustering algorithms is how to decide the lower approximation and boundary region of a cluster. In existing rough-fuzzy clustering [321, 330], the objects are put into lower approximation and boundary region depending upon some user defined threshold values. This decision making step plays a vital role on the performance of the algorithm. In the proposed algorithm, this problem is addressed using the concept of rough hypercuboid approach [312, 526], which implicitly partitions the objects into lower approximation and boundary region of clusters, without the need of any user specified parameter. The cluster prototype is calculated taking the weighted average of the crisp lower approximation and probabilistic boundary. In effect, the proposed algorithm can find overlapping and vaguely defined clusters. The initial prototype selection and stuck in local minima problems of c-means algorithms are efficiently solved by introducing a robust method using fuzzy similarity measure, while the optimum values of different parameters are estimated using a cluster validity index. The efficacy of the proposed algorithm, along with a comparison with existing c-means algorithms, is established on numerous real life data sets with the help of some internal and external validity indices. Some of the results, presented in this chapter, are also reported in [145].

The rest of this chapter is oriented as follows: Section 6.2 briefly describes the necessary concepts of rough hypercuboid approach. In Section 6.3, the proposed RH-IT2FCM algorithm is presented, based on the concept of rough hypercuboid approach and interval type-2 fuzzy c-means. The data sets and cluster validity indices used in this chapter are summarized in Section 6.4. In Section 6.5, the performance is demonstrated on several real life data sets, along with a few case studies and a comparison with other clustering algorithms. The initialization of cluster prototypes and parameter optimization methods are also introduced in this section. Concluding remarks are given in Section 6.6.

6.2 Basics of Rough Hypercuboid Approach

In this section, the basic concepts of rough hypercuboid approach are reported.

Rough Hypercuboid Equivalence Partition Matrix

Let, $\mathbb{U} = \{x_1, \cdots, x_j, \cdots, x_n\}$ be the finite set of n objects, $\mathbb{C} = \{\mathcal{A}_1, \cdots, \mathcal{A}_k, \cdots, \mathcal{A}_m\}$ be the condition attribute set and $\mathbb{D}$ be the decision attribute sets in $\mathbb{U}$. Let $\mathbb{U}/\mathbb{D} = \{\beta_1, \cdots, \beta_i, \cdots, \beta_c\}$ represents c equivalence classes of $\mathbb{U}$, also called information granules, given by the equivalence relation using the decision attribute set $\mathbb{D}$. Similarly, each condition attribute $\mathcal{A}_k \in \mathbb{C}$ generates the c equivalence classes of $\mathbb{U}$, denoted as $\mathbb{U}/\mathcal{A}_k = \{\delta_1, \cdots, \delta_i, \cdots, \delta_c\}$. So, c partitions of $\mathbb{U}$ contain a set of (cn) values $\{\mathrm{h}_{ij}(\mathcal{A}_k)\}$, which can be represented with a $(c \times n)$ matrix $\mathbb{H}(\mathcal{A}_k) = [\mathrm{h}_{ij}(\mathcal{A}_k)]$. The matrix $\mathbb{H}(\mathcal{A}_k)$ is called hypercuboid equivalence partition matrix (HEPM) [312] for the condition attribute $\mathcal{A}_k$ and is represented as

$$\mathbb{H}(\mathcal{A}_k) = \begin{pmatrix} \mathrm{h}_{11}(\mathcal{A}_k) & \mathrm{h}_{12}(\mathcal{A}_k) & \cdots & \mathrm{h}_{1n}(\mathcal{A}_k) \\ \mathrm{h}_{21}(\mathcal{A}_k) & \mathrm{h}_{22}(\mathcal{A}_k) & \cdots & \mathrm{h}_{2n}(\mathcal{A}_k) \\ \cdots & \cdots & \cdots & \cdots \\ \mathrm{h}_{c1}(\mathcal{A}_k) & \mathrm{h}_{c2}(\mathcal{A}_k) & \cdots & \mathrm{h}_{cn}(\mathcal{A}_k) \end{pmatrix} \tag{6.1}$$

$$\text{where } \mathrm{h}_{ij}(\mathcal{A}_k) = \begin{cases} 1 & \text{if } \mathrm{L}_i \leq x_j(\mathcal{A}_k) \leq \mathrm{U}_i \\ 0 & \text{otherwise.} \end{cases} \tag{6.2}$$

Here $\mathrm{h}_{ij}(\mathcal{A}_k) \in \{0, 1\}$ denotes the membership of object x_j for the ith equivalence partition or class β_i. The interval $[\mathrm{L}_i, \mathrm{U}_i]$ is the value range of condition attribute $\mathcal{A}_k$ with respect to class β_i. Following two conditions must be satisfied:

$$1 \leq \sum_{j=1}^{n} \mathrm{h}_{ij}(\mathcal{A}_k) \leq n, \forall i; \quad 1 \leq \sum_{i=1}^{c} \mathrm{h}_{ij}(\mathcal{A}_k) \leq c, \forall j. \tag{6.3}$$

Confusion vector is used to identify the misclassified objects of implicit hypercuboids, constructed using hypercuboid equivalence partition matrix, and is defined as:

$$\mathbb{V}(\mathcal{A}_k) = [\mathrm{v}_1(\mathcal{A}_k), \cdots, \mathrm{v}_j(\mathcal{A}_k), \cdots, \mathrm{v}_n(\mathcal{A}_k)]; \tag{6.4}$$

$$\text{where, } \mathrm{v}_j(\mathcal{A}_k) = \min\{1, \sum_{i=1}^{c} \mathrm{h}_{ij}(\mathcal{A}_k) - 1\}. \tag{6.5}$$

In other words, an object x_j, which is in the lower approximation of any class β_i, will not be in the lower or upper approximations of any other class and $\mathrm{v}_j(\mathcal{A}_k) = 0$. On the contrary, if the object x_j falls in the boundary region of multiple classes, then it must be inside some implicit hypercuboid and $\mathrm{v}_j(\mathcal{A}_k) = 1$. So, the lower approximation and boundary region of the ith class β_i corresponding to the decision attribute set $\mathbb{D}$ can be determined using the hypercuboid equivalence partition matrix and the confusion vector corresponding to the condition attribute $\mathcal{A}_k$. To demonstrate it clearly, Figure 6.1 presents the scatter plots of samples from two classes, namely, Class A and Class B, considering two attributes A_i and A_j.

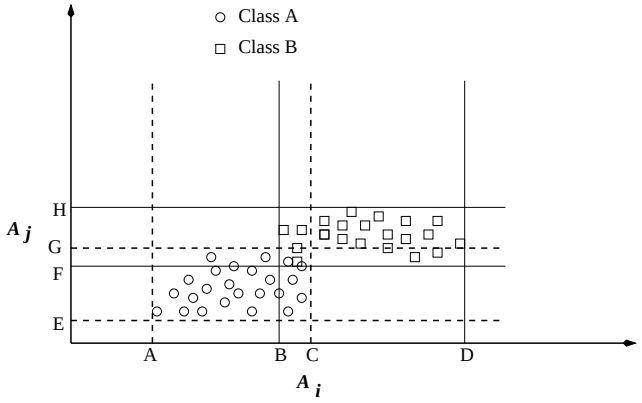


Figure 6.1: Rough hypercuboids in two dimension

Let, $\beta_i \subseteq \mathbb{U}$. A-lower and A-upper approximations of β_i can be constructed to approximate β_i, using only the information contained within $\mathcal{A}_k$:

$$\underline{A}(\beta_i) = \{x_j |\ \mathrm{h}_{ij}(\mathcal{A}_k) = 1 \text{ and } \mathrm{v}_j(\mathcal{A}_k) = 0\}; \tag{6.6}$$

$$\overline{A}(\beta_i) = \{x_j |\ \mathrm{h}_{ij}(\mathcal{A}_k) = 1\}; \tag{6.7}$$

where equivalence relation A is generated from attribute $\mathcal{A}_k$. The boundary region corresponding to β_i is given by

$$B_A(\beta_i) = \overline{A}(\beta_i) \setminus \underline{A}(\beta_i); \tag{6.8}$$

$$\text{that is, } B_A(\beta_i) = \{x_j |\ \mathrm{h}_{ij}(\mathcal{A}_k) = 1 \text{ and } \mathrm{v}_j(\mathcal{A}_k) = 1\}. \tag{6.9}$$

A $c \times n$ hypercuboid equivalence partition matrix $\mathbb{H}(\mathcal{A}_k)$ represents the c-hypercuboid equivalence partitions of the universe generated by an equivalence relation. Each row of $\mathbb{H}(\mathcal{A}_k)$ represents a hypercuboid equivalence partition or class. The ith hypercuboid equivalence partition is represented as

$$\beta_i = \{\mathrm{h}_{i1}(\mathcal{A}_k)/x_1 + \mathrm{h}_{i2}(\mathcal{A}_k)/x_2 + \cdots + \mathrm{h}_{in}(\mathcal{A}_k)/x_n\}. \tag{6.10}$$

An equivalence relation corresponding to a hypercuboid partition generates the equivalence class. Here, "+" represents the union operation. The cardinality of the set, that is, the cardinality of the upper approximation of the class β_i is as follows :

$$|\overline{A}(\beta_i)| = \sum_{j=1}^{n} \mathrm{h}_{ij}(\mathcal{A}_k); \tag{6.11}$$

which can be considered as a natural generalization of crisp set. In a same way, the cardinalities corresponding to lower approximation and boundary region of class β_i may be calculated using these equations:

$$|\underline{A}(\beta_i)| = \sum_{j=1}^{n} \mathrm{h}_{ij}(\mathcal{A}_k) \cap [1 - \mathrm{v}_j(\mathcal{A}_k)]; \tag{6.12}$$

$$|B_A(\beta_i)| = \sum_{j=1}^{n} \mathrm{h}_{ij}(\mathcal{A}_k) \cap \mathrm{v}_j(\mathcal{A}_k). \tag{6.13}$$

Given $< \mathbb{U}, \mathbb{A} >$, $\mathcal{A}_k$ and $\mathcal{A}_l$ are two condition attributes of $\mathbb{A}$. Two $c \times n$ hypercuboid equivalence partition matrices, $\mathbb{H}(\mathcal{A}_k)$ and $\mathbb{H}(\mathcal{A}_l)$, can be used to construct the $c \times n$ hypercuboid equivalence partition matrix for the set of attributes $\{\mathcal{A}_k, \mathcal{A}_l\}$:

$$\mathbb{H}(\{\mathcal{A}_k, \mathcal{A}_l\}) = \mathbb{H}(\mathcal{A}_k) \cap \mathbb{H}(\mathcal{A}_l); \tag{6.14}$$

$$\text{where} \quad \mathrm{h}_{ij}(\{\mathcal{A}_k, \mathcal{A}_l\}) = \mathrm{h}_{ij}(\mathcal{A}_k) \cap \mathrm{h}_{ij}(\mathcal{A}_l). \tag{6.15}$$

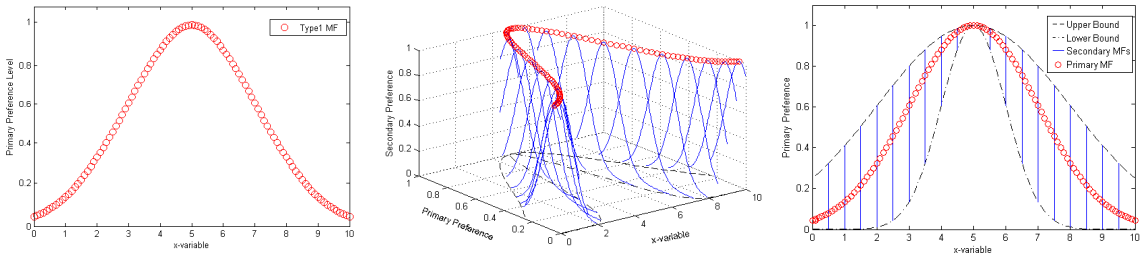
So, the $c \times n$ hypercuboid equivalence partition matrix for the set of condition attributes $\mathbb{C} = \{\mathcal{A}_1, \cdots, \mathcal{A}_k, \cdots, \mathcal{A}_m\}$ is represented as

$$\mathbb{H}(\mathbb{C}) = \bigcap_{\mathcal{A}_k \in \mathbb{C}} \mathbb{H}(\mathcal{A}_k); \quad \text{where} \quad \mathrm{h}_{ij}(\mathbb{C}) = \bigcap_{\mathcal{A}_k \in \mathbb{C}} \mathrm{h}_{ij}(\mathcal{A}_k). \tag{6.16}$$

This resultant hypercuboid equivalence partition matrix $\mathbb{H}(\mathbb{C})$ can be used to construct the lower approximation and boundary region corresponding the whole feature set $\mathbb{C}$.

6.3 Rough Hypercuboid Based IT2 Fuzzy C-Means

The objective function based clustering is the most well known method, where the distance between an object and corresponding cluster prototype needs to be minimized iteratively. In the HCM, all the samples belonging to a cluster contribute equally for taking the decision. If a data set has two identical clusters, C_1 and C_2 (in terms of structure and density), the decision boundary will just be a line exactly perpendicular to the line joining the two cluster centers. The FCM based algorithms work well for overlapping clusters with similar volumes and hyper spherical shapes and densities, that consider the relative distance of the objects across the clusters. So, an object which is distant from a cluster prototype contributes less in center updating than a pattern near the center. The fuzzifier $\acute{m}$ controls the width of decision boundary. However, in real life non hyper-spherical clusters, having different size and densities (e.g. $c_2 > c_1$, where c_1 and c_2 are the radii of the clusters C_1 and C_2, respectively), the performance of the FCM is extremely related with the choice of $\acute{m}$ and must be chosen according to the distribution of the patterns, where improper choice may cause undesirable clusters. If $\acute{m}$ has large value, the estimated cluster center v_1 will deviate from original position and tend to move towards C_2. So, the ideal case will be to set the maximum fuzzy region with wide region to C_1 side and narrow region to C_2 side.



(a) (b) (c)

Figure 6.2: Fuzzy type-1 and type-2 membership functions (a) Gaussian type-1 membership function (b) General type-2 with Gaussian uniform membership function (c) Interval type-2 membership function with fixed secondary grade.

It is clear that the single fuzzifier used in the FCM cannot handle any uncertainty existing in fuzzifier $\acute{m}$. In this regard, Hwang and Rhee [206] used two fuzzifiers, namely, $\acute{m}_1$ and $\acute{m}_2$, with different fuzzy degrees to represent and manage uncertainty, providing an appropriate maximum fuzzy boundary. So, when uncertainty is present in fuzzy memberships, the conventional fuzzy sets can be extended to IT2 fuzzy sets using the two fuzzifiers. This concept is used in the proposed rough hypercuboid based interval type-2 fuzzy c-means (RH-IT2FCM). Fig. 6.2(a) represents a Gaussian T1 fuzzy membership

function. When the secondary grade is present, the corresponding representation is termed as general T2 fuzzy membership function (Fig. 6.2(b)). The IT2 membership function with fixed secondary grade is represented in Fig. 6.2(c). The proposed RH-IT2FCM method uses the concepts of IT2 fuzzy memberships of fuzzy sets, and lower approximation and boundary region of rough sets into c-means algorithm. The rough hypercuboid concept is used to construct the positive and boundary region without the need of any user specified threshold. While the use of probabilistic membership introduced by fuzzy sets allows efficient handling of overlapping partitions in uncertain environment, the rough sets deal with uncertainty, incompleteness, and vagueness in cluster definition.

6.3.1 Lower Approximation and Boundary Region

The RH-IT2FCM algorithm, in each iteration, divides the samples, corresponding to each cluster, into two regions, namely, lower approximation and boundary region, where lower approximation is crisp and boundary region is IT2 fuzzy. This decision making step plays a vital role for the performance of the algorithm. The rough hypercuboid approach is employed in the current study that implicitly divides the samples and put them either in lower approximation or in boundary region of each cluster. A hypercuboid, also called hyperrectangle, is the generalized representation of a rectangle in higher dimensions. It represents the Cartesian product in orthogonal intervals. An m-dimensional hypercuboid is defined in the m-dimensional Euclidean space, where m variables are measured for each sample or object [526]. For real life data set, some implicit hypercuboids are generated from the intersections of the explicit hypercuboids corresponding to different clusters. If any object falls in only one hypercuboid corresponding to a particular cluster, then that object can be put in the lower approximation of that cluster. Otherwise, if it falls in the intersection of two or more hypercuboids, then the object can be put in the boundary region of the clusters corresponding to the hypercuboids.

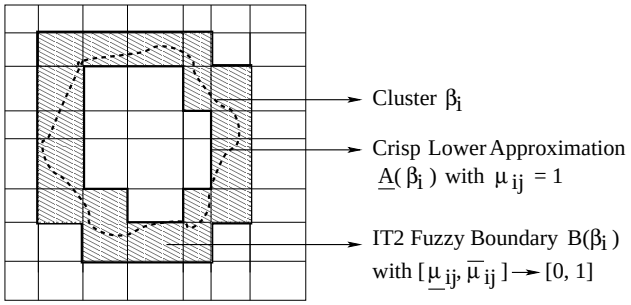


Figure 6.3: RH-IT2FCM: Cluster β_i is represented by crisp lower bound and interval type-2 fuzzy boundary

6.3.2 Objective Function

The RH-IT2FCM algorithm is an objective function based iterative optimization technique, where the sum of the distances between the objects and corresponding cluster prototypes need to be minimized. Let, $X = \{x_1, \cdots, x_j, \cdots, x_n\}$ consists of n objects and $V = \{v_1, \cdots, v_i, \cdots, v_c\}$ be the set of c centroids, where $x_j \in \Re^m$ and $v_i \in \Re^m$. Each cluster β_i is represented using a cluster center v_i, a lower approximation $\underline{A}(\beta_i)$ and a boundary region $B(\beta_i) = \{\overline{A}(\beta_i) \setminus \underline{A}(\beta_i)\}$, where $\overline{A}(\beta_i)$ represents the upper approximation of cluster β_i. Let $d_{ij} = ||x_j - v_i||$ be the distance between jth object x_j and the ith cluster center v_i. The proposed RH-IT2FCM algorithm partitions the set X of objects into c clusters, by minimizing the following objective function:

$$J_{\text{RH-IT2FCM}} = \sum_{i=1}^{c} [w\mathcal{A}_i + (1-w)\mathcal{B}_i] \tag{6.17}$$

$$\text{where } \mathcal{A}_i = \sum_{x_j \in \underline{A}(\beta_i)} d_{ij}^2 \text{ and } \mathcal{B}_i = \sum_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}} d_{ij}^2.$$

The parameters w and $(1-w)$ are used to represent the relative importance of lower approximation and boundary regions. These regions are formed using rough hypercuboid approach of [312]. Here $\acute{m} \in [1, \infty)$ is the probabilistic fuzzifier and $\mu_{ij} \in [0, 1]$ is the probabilistic membership function as similar to that of the RFCM. In case of IT2 fuzzy set, μ_{ij} can be represented by its lower and upper membership degrees of IT2 fuzzy sets, that is, $\mu_{ij} = [\underline{\mu}_{ij}, \overline{\mu}_{ij}]$. Here, fuzzifier $\acute{m}$ can be replaced by $\acute{m}_1$ and $\acute{m}_2$, which correspond to different fuzzy degrees and provide different objective functions. So, the expression of $\mathcal{B}_i$ can be represented as the range $[\underline{\mathcal{B}}_i, \overline{\mathcal{B}}_i]$:

$$\begin{aligned} \mathcal{B}_i = \Big[\Big(& \sum_{\substack{x_j \in B(\beta_i): \\ \rho < c}} (\overline{\mu}_{ij})^{\acute{m}_2} d_{ij}^2 + \sum_{\substack{x_j \in B(\beta_i): \\ \rho \geq c}} (\underline{\mu}_{ij})^{\acute{m}_1} d_{ij}^2 \Big), \\ & \Big(\sum_{\substack{x_j \in B(\beta_i): \\ \rho < c}} (\underline{\mu}_{ij})^{\acute{m}_1} d_{ij}^2 + \sum_{\substack{x_j \in B(\beta_i): \\ \rho \geq c}} (\overline{\mu}_{ij})^{\acute{m}_2} d_{ij}^2 \Big) \Big]; \end{aligned} \tag{6.18}$$

where $\rho = \sum_{k=1}^{c} \frac{d_{ij}}{d_{kj}}$.

In the RH-IT2FCM, a centroid, a crisp lower approximation, and a probabilistic (IT2 fuzzy) boundary (Fig. 6.3) are used to represent a cluster. The fuzziness of resultant partition is affected by the lower approximation. As per the definitions of rough sets [414], if any object $x_j \in \underline{A}(\beta_i)$, then $x_j \notin \underline{A}(\beta_k), \forall k \neq i$, and $x_j \notin B(\beta_i), \forall i$. That is, the object

x_j must be a member of β_i. So, the memberships of any object in lower approximation of a cluster must not be related with other cluster centroids. Here all the objects lying in lower approximation have same influence on the computation of the corresponding centroid for the cluster. So, like the RFCM, the membership of any object in the lower approximation of any cluster is crisp, whereas if $x_j \in B(\beta_i)$, then the object x_j probably belongs to cluster β_i and likely to be a member of another cluster. So, different objects in boundary regions must have different influences on the centroid and cluster updation, and the memberships of the objects should depend on the distances from all cluster centroids. Hence, in the RH-IT2FCM, the membership values of the objects in lower approximation are the same as that of the RFCM, while those in boundary region are identical to that of the IT2FCM.

6.3.3 Membership Function

Minimization of the objective function $J_{\text{RH-IT2FCM}}$, given in (6.17), with respect to μ_{ij}, under the constraint

$$\sum_{i=1}^{c} \mu_{ij} = 1; \tag{6.19}$$

is equivalent to minimizing the following Lagrangian with respect to μ_{ij}:

$$\mathcal{L}(\mu_{ij}, \lambda) = \sum_{i=1}^{c} \mathcal{B}_i + \sum_{x_j \in B(\beta_i)} \lambda\Big(1 - \sum_{i=1}^{c} \mu_{ij}\Big); \tag{6.20}$$

where λ is the Lagrange multiplier. So, $\frac{\partial \mathcal{L}}{\partial \mu_{ij}} = 0$ leads to

$$\lambda = \acute{m}(\mu_{ij})^{\acute{m}-1} d_{ij}^2 \Rightarrow \mu_{ij} = \left(\frac{\lambda}{\acute{m} d_{ij}^2}\right)^{\frac{1}{\acute{m}-1}}. \tag{6.21}$$

Combining (6.19) and (6.21), we get

$$\lambda^{\frac{1}{\acute{m}-1}} = \left[\sum_{k=1}^{c} \left(\frac{1}{\acute{m} d_{kj}^2}\right)^{\frac{1}{\acute{m}-1}}\right]^{-1}. \tag{6.22}$$

So, from (6.21) and (6.22), we get

$$\mu_{ij} = \left[\sum_{k=1}^{c} \left(\frac{d_{ij}^2}{d_{kj}^2}\right)^{\frac{1}{\acute{m}-1}}\right]^{-1}. \tag{6.23}$$

In the RH-IT2FCM, both upper and lower memberships are employed to manage the uncertainties for the fuzzifier $\acute{m}$. The FOU (4.5 of Chapter 4) of fuzzifier $\acute{m}$ is represented by two different fuzzifiers $\acute{m}_1$ and $\acute{m}_2$. The lower (upper) membership $\underline{\mu}_{ij}$ ($\overline{\mu}_{ij}$) is, respectively, the lower (upper) bound of the FOU, which can be derived by solving (6.17) and (6.18) with respect to $\underline{\mu}_{ij}$ and $\overline{\mu}_{ij}$ subject to $\sum_{i=1}^{c} \underline{\mu}_{ij} = 1$ and $\sum_{i=1}^{c} \overline{\mu}_{ij} = 1$ as:

$$\underline{\mu}_{ij} = \begin{cases} \left[\sum_{k=1}^{c} \left(\frac{d_{ij}^2}{d_{kj}^2}\right)^{\frac{1}{\acute{m}_1 - 1}}\right]^{-1} & \text{if } \rho < c \\ \left[\sum_{k=1}^{c} \left(\frac{d_{ij}^2}{d_{kj}^2}\right)^{\frac{1}{\acute{m}_2 - 1}}\right]^{-1} & \text{otherwise} \end{cases}$$

$$\overline{\mu}_{ij} = \begin{cases} \left[\sum_{k=1}^{c} \left(\frac{d_{ij}^2}{d_{kj}^2}\right)^{\frac{1}{\acute{m}_1 - 1}}\right]^{-1} & \text{if } \rho \geq c \\ \left[\sum_{k=1}^{c} \left(\frac{d_{ij}^2}{d_{kj}^2}\right)^{\frac{1}{\acute{m}_2 - 1}}\right]^{-1} & \text{otherwise} \end{cases} \tag{6.24}$$

As the output of the RH-IT2FCM algorithm is an IT2 fuzzy set, it cannot be transformed to crisp set directly by defuzzifier. Type-reduction is needed to reduce the IT2 fuzzy centroid to T1 fuzzy centroid [241]. By using centroid type-reducer, one can get accurate cluster centers and desirable clustering solutions. Then, a centroid defuzzifier generates a crisp center from the type-reduced T1 fuzzy set [241].

6.3.4 Cluster Prototypes

The weighted average of the objects in crisp lower approximation and probabilistic boundary is used to calculate new centroid. The effects of fuzzy membership, lower approximation, and boundary region of rough sets are included in computation of a centroid. The modified centroid calculation for the RH-IT2FCM is obtained by solving (6.17) with respect to v_i and represented as:

$$v_i^{\text{RH-IT2FCM}} = w\mathcal{C}_i + (1-w)\mathcal{D}_i \tag{6.25}$$

$$\text{where } \mathcal{C}_i = \frac{1}{|\underline{A}(\beta_i)|} \sum_{x_j \in \underline{A}(\beta_i)} x_j. \tag{6.26}$$

The term $\mathcal{D}_i$ can be determined by type reduction [538] using Karnik-Mendel (KM) algorithm [241] as follows. First, all objects $x_j \in B(\beta_i)$ are sorted as

$$
\begin{array}{ccccc}
x_{11} & \leqslant & x_{21} & \leqslant \ldots \leqslant & x_{n_i 1} \\
\ldots & & \ldots & & \ldots \\
x_{1m} & \leqslant & x_{2m} & \leqslant \ldots \leqslant & x_{n_i m};
\end{array} \tag{6.27}
$$

where m is the number of features and $n_i = |B(\beta_i)|$ is the number of objects in the boundary region of cluster β_i. The membership μ_{ij} is initialized as $\mu_{ij} = (\underline{\mu}_{ij} + \overline{\mu}_{ij})/2$ and fuzzifier $\acute{m}$ is initialized as $\acute{m} = (\acute{m}_1 + \acute{m}_2)/2$. Then, the ith fuzzy cluster center is calculated as

$$
\acute{v}_i = \frac{\sum\limits_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}} x_j}{\sum\limits_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}}}. \tag{6.28}
$$

Next step is to find the switch point s such that $x_s \leqslant \acute{v}_i \leqslant x_{s+1}$. The minimum value $\acute{v}^L$ and maximum value $\acute{v}^R$ of the ith cluster center can be found out by

$$
\acute{v}_i^L = \min_{\mu_{ij} \in [\underline{\mu}_{ij}, \overline{\mu}_{ij}]} \frac{\sum\limits_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}} x_j}{\sum\limits_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}}} = \frac{\sum\limits_{\substack{j=1 \\ x_j \in B(\beta_i)}}^{s} (\overline{\mu}_{ij})^{\acute{m}} x_j + \sum\limits_{\substack{j=s+1 \\ x_j \in B(\beta_i)}}^{n_i} (\underline{\mu}_{ij})^{\acute{m}} x_j}{\sum\limits_{\substack{j=1 \\ x_j \in B(\beta_i)}}^{s} (\overline{\mu}_{ij})^{\acute{m}} + \sum\limits_{\substack{j=s+1 \\ x_j \in B(\beta_i)}}^{n_i} (\underline{\mu}_{ij})^{\acute{m}}}
$$

$$
\acute{v}_i^R = \max_{\mu_{ij} \in [\underline{\mu}_{ij}, \overline{\mu}_{ij}]} \frac{\sum\limits_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}} x_j}{\sum\limits_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}}} = \frac{\sum\limits_{\substack{j=1 \\ x_j \in B(\beta_i)}}^{s} (\underline{\mu}_{ij})^{\acute{m}} x_j + \sum\limits_{\substack{j=s+1 \\ x_j \in B(\beta_i)}}^{n_i} (\overline{\mu}_{ij})^{\acute{m}} x_j}{\sum\limits_{\substack{j=1 \\ x_j \in B(\beta_i)}}^{s} (\underline{\mu}_{ij})^{\acute{m}} + \sum\limits_{\substack{j=s+1 \\ x_j \in B(\beta_i)}}^{n_i} (\overline{\mu}_{ij})^{\acute{m}}}
$$

$$
\text{where} \quad \acute{m} = \acute{m}_1, \quad \text{if} \quad \begin{cases} \rho < c \quad \text{and} \quad \mu_{ij} = \underline{\mu}_{ij} \\ \rho \geq c \quad \text{and} \quad \mu_{ij} = \overline{\mu}_{ij} \end{cases}
$$

$$\text{and} \quad \acute{m} = \acute{m}_2, \quad \text{if} \quad \begin{cases} \rho < c & \text{and} \quad \mu_{ij} = \overline{\mu}_{ij} \\ \rho \geq c & \text{and} \quad \mu_{ij} = \underline{\mu}_{ij} \end{cases} \tag{6.29}$$

For the computation of $\acute{v}_i^L$, the iteration is stopped if $\acute{v}_i^L = \acute{v}_i$ by setting the minimum value of $\acute{v}_i$ to $\acute{v}_i^L$. Otherwise, setting $\acute{v}_i = \acute{v}_i^L$, the iteration is continued by searching the switch point s again. Similarly, the maximum value of $\acute{v}_i^R$ can be computed. The memberships chosen to update $\acute{v}_i^L$ and $\acute{v}_i^R$ are called μ_{ij}^L and μ_{ij}^R, respectively. The crisp value of $\mathcal{D}_i$ can be calculated by defuzzification of $\acute{v}_i^L$ and $\acute{v}_i^R$ as $\mathcal{D}_i = (\acute{v}_i^L + \acute{v}_i^R)/2$. The new centroid of each cluster is computed using (6.25). The above steps are repeated until no more new assignment is observed.

The choice of the parameter w affects the resultant centroids, and fuzzifiers $\acute{m}_1$ and $\acute{m}_2$ rule their relative influence. The parameter w controls the performance of the RH-IT2FCM. As the lower approximation contains the objects that definitely belong to the cluster, a higher weight w is assigned to them compared to the weight $(1 - w)$ assigned for the objects residing in boundary regions. But the performance of proposed algorithm significantly reduces when the value of w is set nearly equal to 1. In this case, as the boundary regions are ignored, the mobility of the centroids and the objects in the clusters reduces. So, some centroids get stuck in local optimum and cannot change in iterations. Hence, to give the chance to the objects between different clusters and the centroids to shift, the weight is set as $0 < (1 - w) < w < 1$.

6.3.5 RH-IT2FCM Algorithm

The algorithm starts by choosing c objects as initial prototypes of c clusters, and assigning each object x_j to the group having closest centroid. The values of the fuzzifiers $\acute{m}_1$ and $\acute{m}_2$ are selected. Then, for each feature $\mathcal{A}_l \in \mathbb{C}$, the corresponding HEPM $\mathbb{H}(\mathcal{A}_l)$ is constructed according to (6.1), considering initial cluster labels of objects. From all the HEPMs, resultant HEPM $\mathbb{H}(\mathbb{C})$ is constructed using (6.16), considering all features $\mathcal{A}_l \in \mathbb{C}$. The lower approximation $\underline{A}(\beta_i)$ and boundary region $B(\beta_i)$ for each cluster β_i are obtained using (6.6) and (6.9), respectively. If any object $x_j \in \underline{A}(\beta_i)$, then x_j is definitely in cluster β_i and implicitly $x_j \in \overline{A}(\beta_i)$, otherwise $x_j \in \overline{A}(\beta_i)$ and $x_j \in \overline{A}(\beta_k)$. Using the rough hypercuboid approach, all the objects are assigned either in lower approximations or in boundary regions of different clusters, and then the IT2 probabilistic memberships $\underline{\mu}_{ij}$ and $\overline{\mu}_{ij}$ of the objects in the boundary regions are computed according to (6.24). The modified centroids v_i are computed as per (6.25). Karnik-Mendel (KM) algorithm [241] is used to compute the centroid for the objects in boundary region. The above steps are repeated until no more change is observed for the centroids. The major steps of the algorithm are as follows:

1. Set iteration counter $t = 1$. Choose initial prototypes $v_i^{(t)}$, $i = 1, 2, \cdots, c$. Choose the values of fuzzifiers $\acute{m}_1$ and $\acute{m}_2$. Set $\varepsilon > 0$ to a very small value.

2. Assign each object x_j to the group having closest centroid.

3. Construct the HEPM $\mathbb{H}(\mathcal{A}_l)$ for each features $\mathcal{A}_l \in \mathbb{C}$, considering the cluster label of x_j, $\forall j = 1, 2, \cdots, n$.

4. Construct resultant HEPM $\mathbb{H}(\mathbb{C})$, considering all features $\mathcal{A}_l \in \mathbb{C}$ using (6.16).

5. Calculate the lower approximation $\underline{A}(\beta_i)$ and boundary region $B(\beta_i)$ using (6.6) and (6.9). By properties of rough sets, if $x_j \in \underline{A}(\beta_i)$, then $x_j \in \overline{A}(\beta_i)$. If $x_j \in B(\beta_i)$, then $x_j \in B(\beta_k)$, $i \neq k$. Furthermore, x_j is not member of any lower approximation.

6. For the objects residing in boundary regions of c clusters, calculate $\underline{\mu}_{ij}$ and $\overline{\mu}_{ij}$ using (6.24).

7. Compute modified centroid $v_i^{(t+1)}$ using (6.25), where the centroid for the objects in boundary region is calculated using the KM algorithm.

8. Repeat steps 2 to 7 by increasing t, until no more new assignments can be made, that is, $\|v^{(t+1)} - v^{(t)}\| < \varepsilon$.

Type reduction from IT2 to T1 can be done as follows: $\mu_{ij} = (\mu_{ij}^L + \mu_{ij}^R)/2$, where μ_{ij}^L and μ_{ij}^R are the left and right memberships, respectively, calculated using the KM algorithm. Finally, objects are assigned to the clusters using the maximum membership, that is, $x_j \in \beta_i$ if $\mu_{ij} = \max_k\{\mu_{kj}\}$. For a data set with n objects and m attributes, the proposed RH-IT2FCM algorithm has $\mathcal{O}(tcmn)$ time complexity, while the space complexity is $\mathcal{O}((n + c)m)$.

6.3.6 Selection of Initial Cluster Prototypes

One of the major disadvantages of the c-means algorithms is that they may provide locally optimum solutions based on the initial cluster centroids. Due to this limitation, sometimes final centers get stuck in some location and never get the chance to shift to the actual region, wasting huge computing resources. To reduce this weakness, a novel fuzzy similarity based initial prototype selection method is presented here, which helps the clustering algorithm converging to an optimum or near optimum solution. The proposed prototype selection method uses a fuzzy correlation [65] based quantitative measure, defined next. It is used to measure the similarity between pair of objects.

Definition 6.1 *Fuzzy correlation coefficient between two fuzzy objects $\tilde{X}$ and $\tilde{Y}$ is represented as*

$$\tilde{r}_{X,Y} = \frac{\sum_{i=1}^{m}(\tilde{X}_i - \overline{\tilde{X}})(\tilde{Y}_i - \overline{\tilde{Y}})}{\sqrt{\sum_{i=1}^{m}(\tilde{X}_i - \overline{\tilde{X}})^2}\sqrt{\sum_{i=1}^{m}(\tilde{Y}_i - \overline{\tilde{Y}})^2}}; \tag{6.30}$$

where $\overline{\tilde{X}}$ and $\overline{\tilde{Y}}$ represent the averages over all attributes. Multiple fuzzy numbers are involved in the above equation. So, Zadeh's extension principle [541] can be used to find out the membership function. Using the extension principle, the membership function of $\tilde{r}_{X,Y}$ can be defined as

$$\mu_{\tilde{r}_{X,Y}}(r) = \sup_{X,Y} \min\{\mu_{\tilde{X}_i}(x_i), \mu_{\tilde{Y}_i}(y_i)\}, \forall i | r = r_{X,Y} \tag{6.31}$$

where $r_{X,Y}$ is defined as

$$r_{X,Y} = \frac{\sum_{i=1}^{m}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{m}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{m}(Y_i - \overline{Y})^2}} \tag{6.32}$$

$\tilde{r}_{X,Y}$ can be found out by using α-cuts [65]. Let $(\tilde{r}_{X,Y})_\alpha$ be the fuzzy correlation coefficient for each α level. As every fuzzy object has the membership in both x and y directions, if we take α-cut on the membership function, two intervals are generated in both of the directions for each observation. The α-cuts for $\tilde{X}_i$ and $\tilde{Y}_i$ are denoted as

$$\begin{aligned}(X_i)_\alpha &= [(X_i)_\alpha^L, (X_i)_\alpha^U] \\ &= [\min_x\{x_i \in X | \mu_{\tilde{X}_i}(x_i) \geq \alpha\}, \max_x\{x_i \in X | \mu_{\tilde{X}_i}(x_i) \geq \alpha\}];\end{aligned} \tag{6.33}$$

$$\begin{aligned}(Y_i)_\alpha &= [(Y_i)_\alpha^L, (Y_i)_\alpha^U] \\ &= [\min_y\{y_i \in Y | \mu_{\tilde{Y}_i}(y_i) \geq \alpha\}, \max_y\{y_i \in Y | \mu_{\tilde{Y}_i}(y_i) \geq \alpha\}];\end{aligned} \tag{6.34}$$

where X and Y are the crisp universal sets, corresponding to $\tilde{X}$ and $\tilde{Y}$, respectively.

So, the problem of finding the membership function $\mu_{\tilde{r}_{X,Y}}$ reduces in finding the lower and upper bounds of the α-cut of $\tilde{r}_{X,Y}$ by solving the following pair of equations:

$$(r_{X,Y})_\alpha^L = \min \frac{\sum_{i=1}^{m}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{m}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{m}(y_i - \overline{y})^2}} \tag{6.35}$$

$$(r_{X,Y})_\alpha^U = \max \frac{\sum_{i=1}^{m}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{m}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{m}(y_i - \overline{y})^2}} \tag{6.36}$$

$$\text{such that } \; (X_i)_\alpha^L \leq x_i \leq (X_i)_\alpha^U; \quad (Y_i)_\alpha^L \leq y_i \leq (Y_i)_\alpha^U.$$

This is a problem with two non-linear equations with bounded constrains. The left shape function $L(r)$ and the right shape function $R(r)$ can be constructed by taking the inverse of $(r_{X,Y})_\alpha^L$ and $(r_{X,Y})_\alpha^U$, respectively, which give the membership function $\mu_{\tilde{r}_{X,Y}}$ as follows:

$$\mu_{\tilde{r}_{X,Y}} = \begin{cases} L(r) & \text{if } (r_{X,Y})_{\alpha=0}^L \leq r \leq (r_{X,Y})_{\alpha=1}^L \\ 1 & \text{if } (r_{X,Y})_{\alpha=1}^L \leq r \leq (r_{X,Y})_{\alpha=1}^U \\ R(r) & \text{if } (r_{X,Y})_{\alpha=1}^U \leq r \leq (r_{X,Y})_{\alpha=0}^U \end{cases} \tag{6.37}$$

One way of solving is to gather the numerical solutions for $(r_{X,Y})_\alpha^L$ and $(r_{X,Y})_\alpha^U$ at different α levels and approximate the shape of $L(r)$ and $R(r)$. The fuzzy correlation coefficient ranges from [-1,1]. The closer the absolute value to 1, the stronger is the correlation between X and Y.

The fuzzy correlation $\tilde{r}_{x_i,x_j}$ can be calculated using (6.30) between two fuzzy objects x_i and x_j. Let, $Sim(x_i, x_j) = |\tilde{r}_{x_i,x_j}|$. Since the value of $Sim(x_i, x_j)$ decreases as dissimilarity between objects increases and ranges between zero (in the limit) and one (when $x_i = x_j$), it has a ready interpretation as a similarity measure. The Sim is used to represent the similarity between pairs of objects. If two objects are dissimilar, the Sim among them will be small. A large value of Sim between two objects asserts that they are similar to each other. If two objects are same, the Sim between them is maximum. It satisfies the following conditions:

1. $Sim(x_i, x_i) = 1$
2. $Sim(x_i, x_j) = Sim(x_j, x_i) > 0$

3. If $x_i \leq x_j \leq x_k$ or $x_i \geq x_j \geq x_k$, $\forall i, j, k$, then $Sim(x_i, x_k) \leq Sim(x_j, x_k)$.

Based on the concept of fuzzy similarity, an algorithm is described next to select the initial prototypes. The basic steps of the algorithm are as follows:

1. For each object x_i, calculate $Sim(x_i, x_j)$ between itself and the object x_j, $\forall j = 1, 2 \ldots n$.

2. Calculate the similarity between objects x_i and x_j

$$\mathcal{S}(x_i, x_j) = \begin{cases} 1 & \text{if } Sim(x_i, x_j) > \lambda_1 \\ 0 & \text{otherwise} \end{cases} \tag{6.38}$$

   where $0.5 \leq \lambda_1 \leq 1$.

3. For each object x_i, calculate total number of similar objects of x_i as

$$\mathcal{N}(x_i) = \sum_{j=1}^{n} \mathcal{S}(x_i, x_j). \tag{6.39}$$

4. Sort n objects according to the respective values of $\mathcal{N}(x_i)$ so that $\mathcal{N}(x_1) > \mathcal{N}(x_2) > \cdots > \mathcal{N}(x_n)$.

5. If $\mathcal{N}(x_i) > \mathcal{N}(x_j)$ and

$$Sim(x_i, x_j) > \lambda_2 \tag{6.40}$$

   where $0.5 \leq \lambda_2 \leq 1$, then discard x_j, which cannot be considered as an initial cluster prototype. This process results in a reduced set of objects, which can be considered as the initial cluster prototypes.

Finally, first c objects of the reduced set are considered as potential initial centers. The motivation behind introducing this prototype initialization method is to find out the dense regions from the data set. These dense regions help us to find the natural groups out of the data set. Hence, the overall concept is data dependent.

6.4 Cluster Validity Indices and Data Sets Used

This section presents a brief description of the cluster validity indices and data sets used.

6.4.1 Cluster Validity Indices

Four internal cluster validity indices, namely, Silhouette index [446], DB index [50], Dunn index [50], and Xie-Beni index [539], and eight external cluster validity indices, namely,

precision [424], sensitivity [424], specificity [424], accuracy [424], Jaccard Coefficient [424], Dice Coefficient [104], G Measure [275], and Matthews Correlation Coefficient [339] are used to evaluate the performance of different clustering methods with respect to several real life data sets. The quantitative indices are discussed in Appendix A.

6.4.2 Description of Data Sets

In this chapter, publicly available eight real life data sets are used to compare the performance of different clustering methods, which are downloaded from *UCI Machine Learning Repository* (https://archive.ics.uci.edu/ml/datasets.html). The descriptions are as follows.

6.4.2.1 Glass

The study of classification of types of glass was motivated by criminological investigation, to determine whether the glass was a type of "float" glass or not. The data set contains 214 examples, having 9 attributes and 6 types of glasses.

6.4.2.2 Hepatitis

It contains hepatitis data. 155 instances having 19 mixed types of attributes are present. 2 classes are there in the dataset.

6.4.2.3 Wine

These data are the results of a chemical analysis of wines grown in the same region in Italy, but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines for 178 samples.

6.4.2.4 Wisconsin Breast Cancer (WBC)

This breast cancer database was obtained from the University of Wisconsin Hospitals and contains 699 instances about 458 cases of benign cancer and 241 cases of malignant cancer. 10 attributes are present in the data set.

6.4.2.5 Soybean-Small

It is a small subset of the original soybean database containing 47 samples and 35 normalized attributes. There are 4 classes with 10, 10, 10, and 17 samples respectively.

6.4.2.6 Dermatology

The differential diagnosis of erythemato-squamous diseases is a real problem in dermatology. This database contains 366 instances and 34 attributes, 33 of which are linear valued and rest one is nominal. 6 different classes are present in the data set.

6.4.2.7 Echocardiogram

All the patients suffered heart attacks at some point in the past. Some are still alive and some are not. The survival and still-alive variables, when taken together, indicate whether a patient survived for at least one year following the heart attack. It contains 131 instances, 10 mixed attributes, and 2 classes, dead and alive.

6.4.2.8 Ecoli

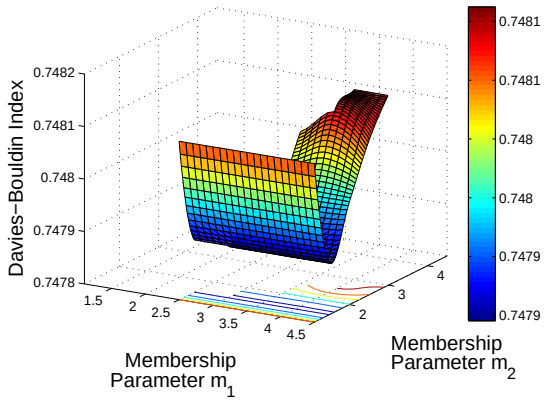
Ecoli data set contains 336 instances representing non-numeric localization sites of protein. 8 predictive attributes are present, along with an accession number for the SWISS-PROT database. There are 8 classes which represent the localization sites.

6.5 Experimental Results and Discussion

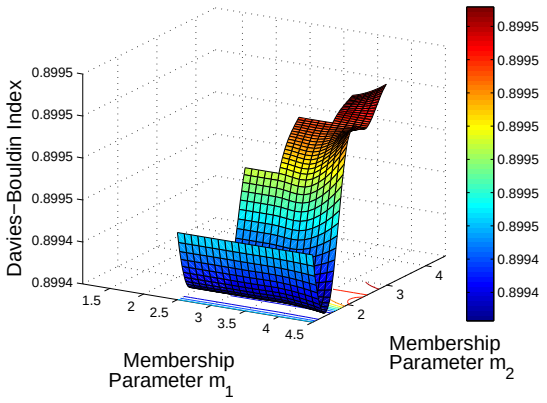
The performance of the proposed RH-IT2FCM algorithm is compared extensively with that of different c-means algorithms on eight real life data sets mentioned in the Section 6.4.2. The performance of the proposed algorithm is compared with that of hard c-means (HCM), fuzzy c-means (FCM) [48], possibilistic c-means (PCM) [261], fuzzy-possibilistic c-means (FPCM) [383], rough c-means (RCM) [289], rough-fuzzy c-means (RFCM) [321], rough-possibilistic c-means (RPCM) [321], rough-fuzzy-possibilistic c-means (RFPCM) [321], kernel based HCM (KHCM) [560], kernel based FCM (KFCM) [560], kernel based PCM (KPCM) [560], kernel based FPCM (KFPCM) [560], rough hypercuboid based FCM (RHFCM), and interval type-2 FCM (IT2FCM) [283, 345]. All the algorithms are implemented in C language and executed in LINUX environment in a machine with Pentium Core i5, 2.4 GHz, 3 MB 12-way SA L3 cache, and 4 GB DDR3 RAM. The values of fuzzifiers $\acute{m}_1$ and $\acute{m}_2$ are varied taking the values from the set {1.1, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5}, for all values of $\acute{m}_2 \geq \acute{m}_1$. The value of weight parameter w for rough-fuzzy clustering is taken as 0.99 as suggested in [321].

6.5.1 Optimum Values of Different Parameters

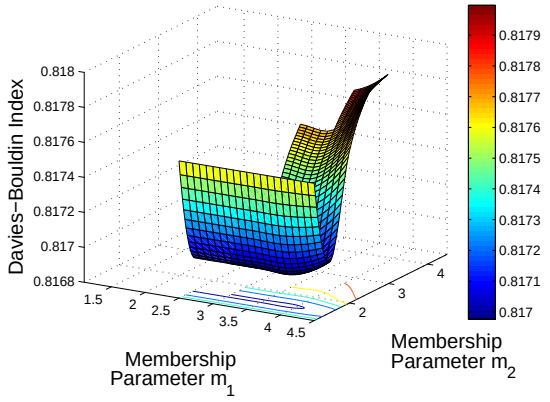
The parameters $\acute{m}_1$ and $\acute{m}_2$ ($\acute{m}_2 \geq \acute{m}_1$) of RH-IT2FCM control the shape of both of the membership functions, giving the lower membership $\underline{\mu}_{ij}$ and upper membership $\overline{\mu}_{ij}$.



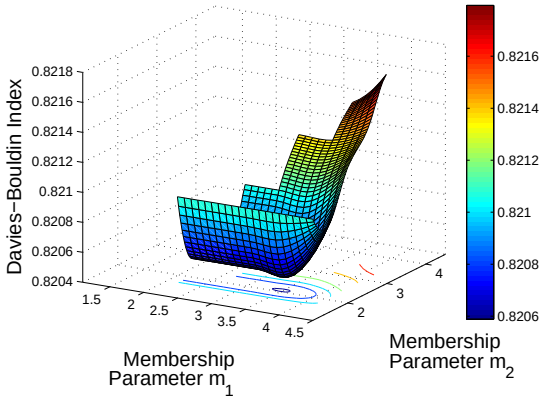
(a) Glass ($\lambda_1^\star = 0.94, \lambda_2^\star = 0.92$)



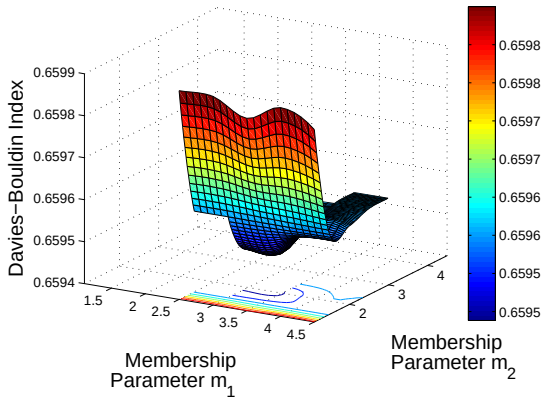
(b) Hepatitis ($\lambda_1^\star = 0.96, \lambda_2^\star = 0.86$)



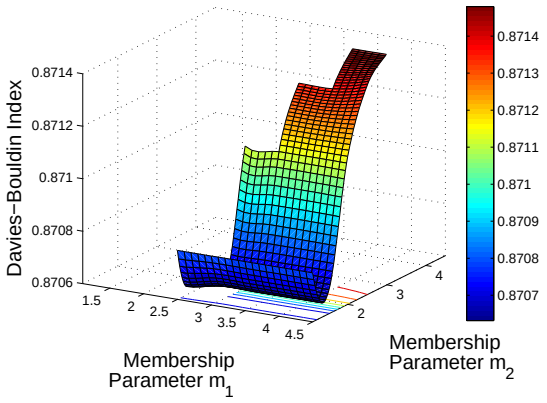
(c) Wine ($\lambda_1^\star = 0.96, \lambda_2^\star = 0.88$)



(d) WBC ($\lambda_1^\star = 0.92, \lambda_2^\star = 0.94$)

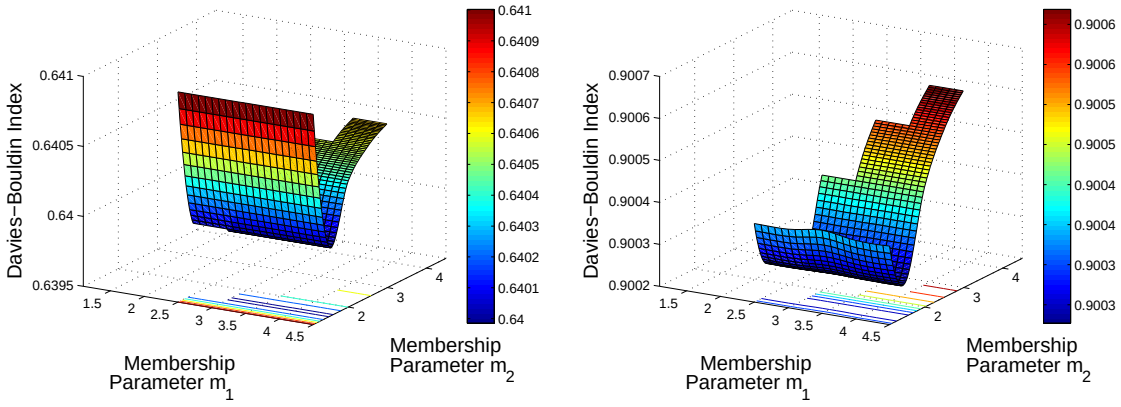


(e) Soybean-Small ($\lambda_1^\star = 0.96, \lambda_2^\star = 0.86$)



(f) Dermatology ($\lambda_1^\star = 0.96, \lambda_2^\star = 0.82$)

Figure 6.4: Variation of DB index for different values of $\acute{m}_1$ and $\acute{m}_2$



(a) Echocardiogram ($\lambda_1^\star = 0.94, \lambda_2^\star = 0.94$)

(b) Ecoli ($\lambda_1^\star = 0.94, \lambda_2^\star = 0.84$)

Figure 6.5: Variation of DB index for different values of $\acute{m}_1$ and $\acute{m}_2$

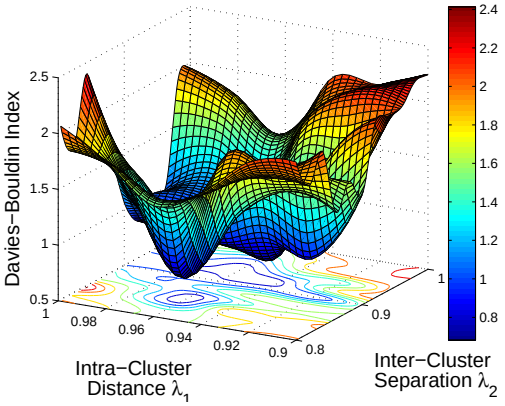
The membership functions can be altered varying $\acute{m}_1$ and $\acute{m}_2$, keeping the values of other parameters fixed. The thresholds λ_1 and λ_2 in (6.38) and (6.40) also have a significant role for the selection of initial cluster prototypes for different clusters. They control the dissimilarity between the initial prototypes. Hence, they have direct influences on the performance of proposed algorithm as well as other c-means algorithms.

The Davies-Bouldin (DB) index [446] is used to find out the optimum values of two fuzzy parameters, namely, $\acute{m}_1$ and $\acute{m}_2$, and two thresholds, namely, λ_1 and λ_2, for different data sets. The DB index is defined as the ratio of sum of within-cluster scatter to between-cluster separation, and is given by

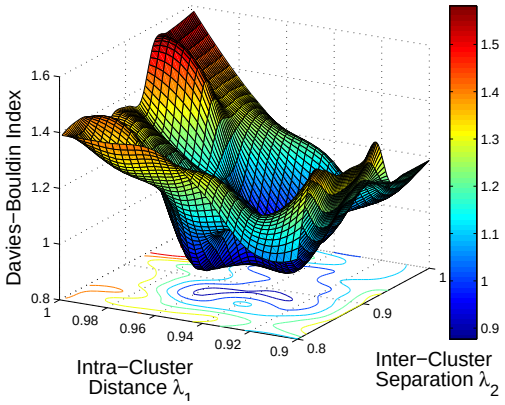
$$\text{DB} = \frac{1}{c}\sum_{i=1}^{c} \max_{i \neq k} \left\{ \frac{S(v_i) + S(v_k)}{d(v_i, v_k)} \right\} \tag{6.41}$$

for $1 \leq i, k \leq c$. The DB index minimizes the within-cluster difference $S(v_i)$ and maximizes the between-cluster distance $d(v_i, v_k)$ for a given data set having c clusters. A good clustering algorithm should provide low DB index.

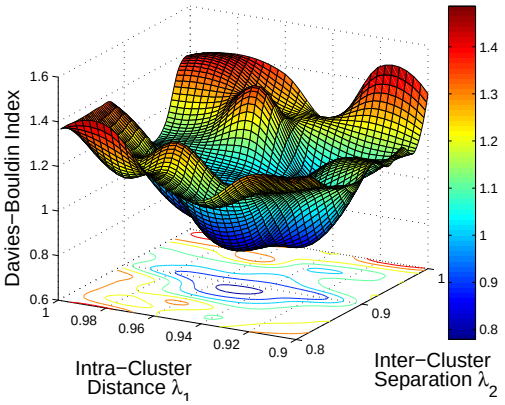
For eight data sets, the values of $\acute{m}_1$ and $\acute{m}_2$ are varied taking the values from the set $\{1.1, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5\}$, for all values of $\acute{m}_2 \geq \acute{m}_1$. The variation of DB index is represented in Fig. 6.4 and 6.5 corresponding to different values of $\acute{m}_1$ and $\acute{m}_2$, considering $w = 0.99$. The results reported in Fig. 6.4 and 6.5 show that as the parameters $\acute{m}_1$ and $\acute{m}_2$ increase, the DB index value decreases and reaches to its minimum value at the specific values of $\acute{m}_1^\star$ and $\acute{m}_2^\star$. After that, the DB index value increases with the increase of $\acute{m}_1$ and $\acute{m}_2$. Similarly, the value of λ_1 is varied from 0.90 to 1.0, while λ_2 is varied from 0.80 to 1.0. The variation of DB index for different values of λ_1 and λ_2 on eight data sets is shown in Fig. 6.6 and 6.7. The results reported in Fig. 6.6 and 6.7 show that as the



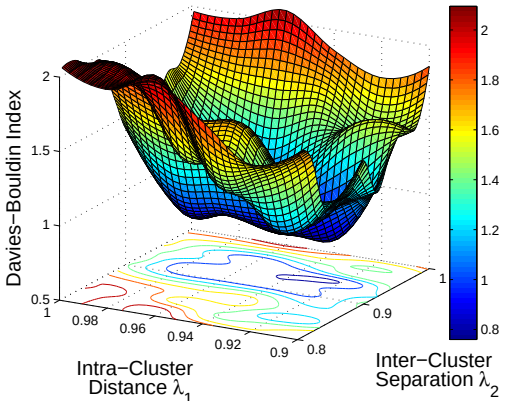
(a) Glass ($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)



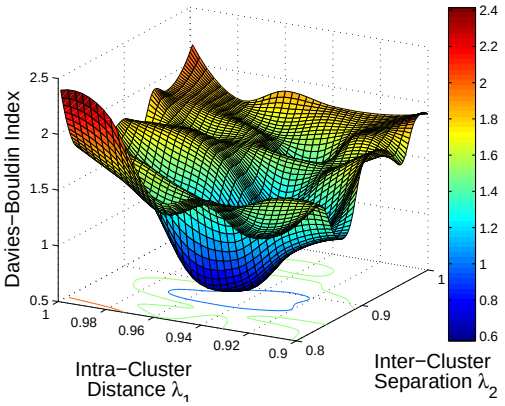
(b) Hepatitis ($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)



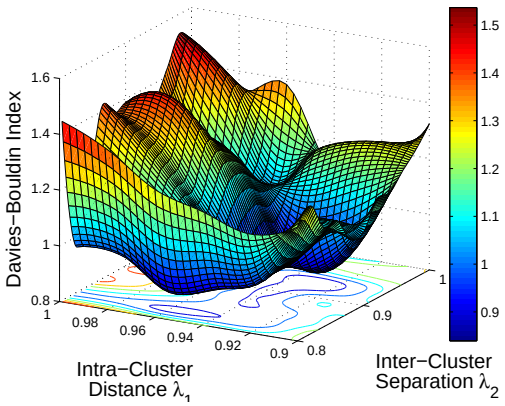
(c) Wine ($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)



(d) WBC ($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)

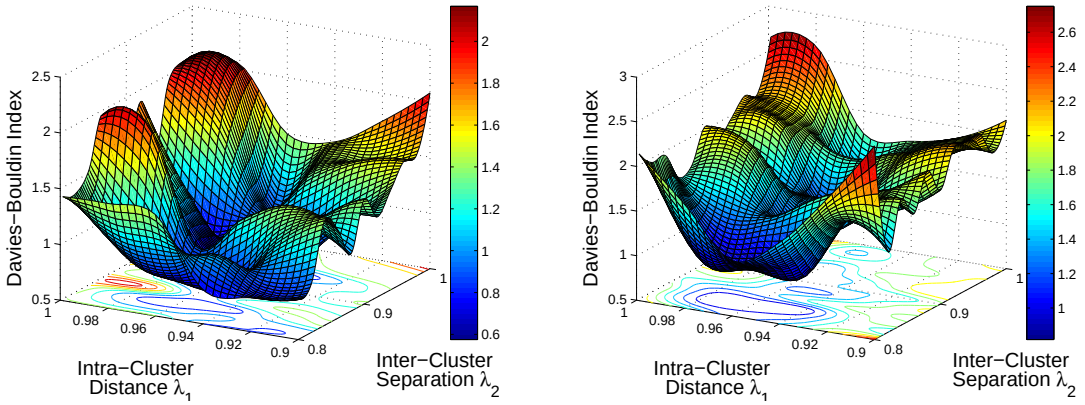


(e) Soybean-Small ($\acute{m}_1^\star = 1.1, \acute{m}_2^\star = 2.0$)



(f) Dermatology ($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)

Figure 6.6: Variation of DB index for different values of λ_1 and λ_2



(a) Echocardiogram($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$) (b) Ecoli ($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)

Figure 6.7: Variation of DB index for different values of λ_1 and λ_2

thresholds λ_1 and λ_2 increase, the DB index value decreases and reaches to its minimum value at the specific values of $\lambda_1^\star$ and $\lambda_2^\star$. Then, the DB index value increases when the values of λ_1 and λ_2 are increased. Hence, the optimum values of $\acute{m}_1$, $\acute{m}_2$, λ_1, and λ_2 for each data set are obtained using the following relation:

$$[\acute{m}_1^\star, \acute{m}_2^\star, \lambda_1^\star, \lambda_2^\star] = \arg \min_{\acute{m}_1, \acute{m}_2, \lambda_1, \lambda_2} \{\text{DB}\}. \tag{6.42}$$

The optimum values of $\acute{m}_1^\star$, $\acute{m}_2^\star$, $\lambda_1^\star$, and $\lambda_2^\star$, obtained using (6.42), are {1.5, 2.0, 0.94, 0.92}, {1.5, 2.0, 0.96, 0.86}, {1.5, 2.0, 0.96, 0.88}, {1.5, 2.0, 0.92, 0.94}, {1.1, 2.0, 0.96, 0.86}, {1.5, 2.0, 0.96, 0.82}, {1.5, 2.0, 0.94, 0.94}, and {1.5, 2.0, 0.94, 0.84}, respectively, for Glass, Hepatitis, Wine, Wisconsin Breast Cancer (WBC), Soybean-Small, Dermatology, Echocardiogram, and Ecoli data sets.

In this regard, it should be noticed that the proposed RH-IT2FCM algorithm attains best performance at $\acute{m}_1 = \acute{m}_1^\star$, $\acute{m}_2 = \acute{m}_2^\star$, $\lambda_1 = \lambda_1^\star$, and $\lambda_2 = \lambda_2^\star$ in 27 cases, out of total 32 cases, with respect to different cluster validity indices. However, the Dunn index provides the maximum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.94, \lambda_2 = 0.88$ for Wine data, at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.96, \lambda_2 = 0.86$ for Dermatology data, and Xie-Beni index attains its minimum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.94, \lambda_2 = 0.88$ for Wine data, at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.94, \lambda_2 = 0.94$ for Wisconsin Breast Cancer data, and at $\lambda_1 = 0.94, \lambda_2 = 0.86$ for Ecoli data. Also, all other clustering algorithms achieve their best performance at $\acute{m}_1 = \acute{m}_1^\star$, $\acute{m}_2 = \acute{m}_2^\star$, $\lambda_1 = \lambda_1^\star$ and $\lambda_2 = \lambda_2^\star$ in 309 cases, out of total 384 cases, irrespective of the cluster validity indices used. Hence, all the results reported above establish the importance of proposed parameter optimization technique, irrespective of the clustering algorithms compared.

6.5.2 Effectiveness of Proposed Initialization Method

Tables 6.1, 6.3, 6.5, 6.7, 6.9, 6.11, 6.13 and 6.15 provide the comparative results of different c-means algorithms with random centroid initialization and the proposed centroid initialization method discussed in Section 6.3.6 for eight data sets. For each c-means algorithm, the results are presented for the optimum values of λ_1 and λ_2. For all c-means algorithms, the performance is significantly improved in most of the cases, if the proposed initialization method is used instead of random initialization. Out of total 384 cases, the proposed initialization method provides significantly better results in 326 cases than the random initialization method. The bold-marked results, reported in Tables 6.1, 6.3, 6.5, 6.7, 6.9, 6.11, 6.13 and 6.15, also prove that the proposed RH-IT2FCM algorithm with proposed initialization method provides better performance than any existing c-means algorithm in almost all the cases, irrespective of initialization methods used.

Moreover, it is also observed that the traditional c-means algorithms with the proposed initialization method provide better results than the proposed RH-IT2FCM algorithm with random initialization method in some cases, in terms of different cluster validity indices used. There are 38, 21, 19, and 19 such cases, in terms of Silhouette index, DB index, Dunn index, and Xie-Beni index, respectively, out of total 96 cases each, where the traditional c-means algorithms with the proposed initialization method provide better results than the RH-IT2FCM algorithm with random initialization method. The proposed initialization method, with optimum parameter values, also provides the best performance in 80.47% cases, where the best performance indicates the best value when initialized by the proposed initialization method introduced in Section 6.3.6 for all possible parameter variations. All the results reported here establish the importance of the proposed prototype initialization algorithm over random initialization method.

6.5.3 Relevance of Rough Hypercuboid Approach

To prove the superiority of the rough hypercuboid approach over traditional rough sets, extensive experimentation is carried out on eight real life data sets. Tables 6.1, 6.3, 6.5, 6.7, 6.9, 6.11, 6.13 and 6.15 provide the results for optimum values of λ_1 and λ_2 with respect to four internal cluster validity indices. Out of 32 comparisons, in 30 cases, the RHFCM algorithm with rough hypercuboid approach performs better than any other methods, in lesser or comparable time, except IT2FCM and RH-IT2FCM. Also, the RHFCM algorithm achieves better results in each and every cases than the RFCM algorithm with traditional rough sets. Tables 6.2, 6.4, 6.6, 6.8, 6.10, 6.12, 6.14 and 6.16 provide the results for optimum values of λ_1 and λ_2 with respect to eight external cluster validity indices. Out of 64 comparisons, in 24 cases, the RHFCM algorithm with rough hypercuboid approach performs better than any other algorithms, except IT2FCM and RH-IT2FCM. Also, out

of 64 cases, the RHFCM algorithm achieves better result in 57 cases than the RFCM algorithm with traditional rough sets.

The best performance of the RHFCM over the RFCM and other rough clustering algorithms, in terms of several internal and external cluster validity indices, is achieved due to the fact that the rough hypercuboid approach, used in the RHFCM, can find the lower approximation and boundary region of a cluster implicitly without using any user defined threshold as done in the RFCM algorithm.

6.5.4 Importance of Interval Type-2 Fuzzy Set

To show the importance of the IT2 rough-fuzzy approach over the traditional rough-fuzzy approach, the results of Tables 6.1, 6.3, 6.5, 6.7, 6.9, 6.11, 6.13 and 6.15 are analyzed with respect to internal cluster validity indices. Out of total 32 cases, the RH-IT2FCM algorithm with IT2 fuzzy set performs better than the RHFCM algorithm with T1 fuzzy set in 31 cases with lesser or comparable time. Tables 6.2, 6.4, 6.6, 6.8, 6.10, 6.12, 6.14 and 6.16 provide the comparative results with respect to external cluster validity indices. Out of 64 comparisons, in 63 cases, the RH-IT2FCM algorithm performs better than the RHFCM algorithm. The best performance of the RH-IT2FCM over the RHFCM, in terms of different cluster validity indices, is achieved because the IT2 fuzzy set helps to manage the uncertainty present in the fuzzy membership function.

6.5.5 Performance of Different Clustering Algorithms

Finally, the performance of the proposed c-means algorithm is compared with that of rough-fuzzy c-means (RFCM) [321], interval type-2 fuzzy c-means (IT2FCM) [206], rough-hypercuboid based fuzzy c-means (RHFCM), and other existing c-means algorithms. Results and corresponding discussions, on eight real life data sets, are presented in Tables 6.1, 6.3, 6.5, 6.7, 6.9, 6.11, 6.13 and 6.15 with respect to internal cluster validity indices, while in Tables 6.2, 6.4, 6.6, 6.8, 6.10, 6.12, 6.14 and 6.16 with respect to external cluster validity indices. All the results correspond to optimum values of λ_1 and λ_2. The bold values in these tables signify the best values. Out of total 32 cases, in 30 cases, the RH-IT2FCM algorithm with the proposed initialization method provides better results than all other algorithms, with respect to internal cluster validity indices. On the other hand, the RH-IT2FCM algorithm achieves better results than other methods in 59 cases, out of total 64 comparisons, with respect to external cluster validity indices.

From all the results reported in Tables 6.1 – 6.16, the following conclusions can be drawn:

1. The proposed clustering algorithm is better than any other fuzzy, rough-fuzzy and

Table 6.1: Performance of Different Clustering Algorithms for Glass Dataset using Internal Cluster Validity Indices

Different Algorithms	Silhouette Index			DB Index			Dunn Index			Xie-Beni Index		
	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best
HCM	0.235736	0.310473	0.310896	1.980537	1.213282	1.213282	0.153467	0.168897	0.168897	0.728707	0.250059	0.250059
FCM	0.310456	0.293929	0.293929	5.856671	5.618930	4.023843	0.041651	0.043725	0.043725	0.557441	0.394314	0.394314
PCM	-	-	-	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-	-	-	-
RCM	0.235736	0.310473	0.310473	1.980537	1.213282	1.213282	0.163424	0.168897	0.168897	0.785913	0.250059	0.250059
RFCM	0.240935	0.242844	0.242844	3.009344	1.744338	1.744338	0.196175	0.186706	0.196175	0.786554	0.335330	0.334350
RPCM	0.233765	0.240700	0.240700	1.347800	0.933801	0.933801	0.091737	0.128600	0.128600	0.742241	0.412367	0.390846
RFPCM	0.234132	0.239097	0.239097	1.082774	0.983218	0.983218	0.133890	0.197640	0.209845	0.469041	0.274675	0.274675
KHCM	0.235288	0.327444	0.328432	1.965152	1.074624	1.074624	0.218621	0.230084	0.230084	0.484824	0.261985	0.261985
KFCM	0.314113	0.291144	0.291144	7.476635	6.593579	5.109845	0.028936	0.035617	0.035617	0.740068	0.404054	0.404054
KPCM	-	-	-	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-	-	-	-
IT2FCM	0.367162	**0.370813**	**0.370813**	2.686164	2.577472	2.348756	0.563758	0.216541	0.216541	0.448346	0.551127	0.551127
RHFCM	**0.396136**	0.282803	0.297865	1.297406	0.931742	0.931742	0.856752	0.795429	0.879843	0.340919	**0.232347**	0.232347
RH-IT2FCM	0.360405	0.366560	0.366560	**0.976500**	**0.747898**	**0.747898**	**0.846884**	**1.011817**	**1.011817**	**0.340919**	0.235142	**0.232347**

Table 6.2: Performance of Different Algorithms for Glass Dataset using External Cluster Validity Indices

Different Algorithms	Precision	Sensitivity	Specificity	Accuracy	Jaccard Coefficient	Dice Coefficient	G Measure	Matthews Corr Coeff	Time (msec)
HCM	0.370074	0.399442	0.876769	0.816199	0.237775	0.384198	0.384478	0.260748	56
FCM	0.398862	0.434217	0.877765	0.800623	0.262458	0.415789	0.416164	0.287251	178
PCM	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-
RCM	0.335334	0.335418	0.861854	0.778816	0.201473	0.335376	0.335376	0.193955	59
RFCM	0.370157	0.407499	0.868287	0.799065	0.240642	0.387931	0.388379	0.249593	76
RPCM	0.300039	0.208127	0.842269	0.767913	0.140102	0.245771	0.249892	0.085378	80
RFPCM	0.370370	0.253617	0.864245	0.813520	0.177212	0.301071	0.306483	0.161895	88
KHCM	0.353287	0.319389	0.865009	0.799065	0.201550	0.335484	0.335911	0.200436	120
KFCM	0.365265	0.366023	0.885952	0.809969	0.223724	0.365644	0.365644	0.246097	297
KPCM	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-
IT2FCM	0.409204	**0.463126**	0.875091	0.800623	0.277545	0.434498	0.435331	0.295875	359
RHFCM	0.413290	0.437394	0.887096	0.841121	0.269842	0.425001	0.425171	0.314407	47
RH-IT2FCM	**0.426980**	0.451680	**0.894342**	**0.850467**	**0.281216**	**0.438983**	**0.439156**	**0.335526**	117

Table 6.3: Performance of Different Clustering Algorithms for Hepatitis Dataset using Internal Cluster Validity Indices

Different Algorithms	Silhouette Index			DB Index			Dunn Index			Xie-Beni Index		
	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best
HCM	0.258142	0.258337	0.258337	2.540374	2.539697	2.539697	0.611053	0.621092	0.621092	1.182734	1.171075	1.171075
FCM	0.249120	0.249120	0.249120	2.540123	2.530887	2.530887	0.496727	0.610981	0.610981	1.278374	1.018323	1.018323
PCM	-	-	-	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-	-	-	-
RCM	0.258337	0.258337	0.258337	2.540374	2.539697	2.539697	0.641050	0.621092	0.621092	1.152279	1.171075	1.148734
RFCM	0.266864	0.186720	0.186720	2.540123	2.530887	2.530887	0.596736	0.610981	0.610981	1.127565	1.009148	1.009148
RPCM	0.283477	0.283477	0.283477	2.111213	2.035029	2.035029	0.583539	0.605755	0.605755	1.433470	1.381162	1.320745
RFPCM	0.249761	0.249761	0.258337	2.002390	1.931518	1.930172	0.749852	0.777733	0.777733	0.977605	0.943047	0.943047
KHCM	0.258142	0.258337	0.258337	2.569575	2.554875	2.554875	0.615016	0.617402	0.617402	1.190136	1.178074	1.178074
KFCM	0.191989	0.196492	0.213764	2.558432	2.538423	2.538423	0.617232	0.621092	0.622066	1.124543	1.032743	1.030073
KPCM	-	-	-	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-	-	-	-
IT2FCM	0.258653	0.287953	0.287953	2.995113	1.309513	1.309513	0.415190	1.198035	1.198035	0.645347	0.523572	0.523572
RHFCM	0.345581	0.401031	0.401031	0.917255	0.899528	0.899528	1.653956	2.001056	2.001056	0.600844	0.402371	0.402371
RH-IT2FCM	**0.345581**	**0.401031**	**0.401031**	**0.917255**	**0.899442**	**0.899442**	**1.653956**	**2.001251**	**2.001251**	**0.600844**	**0.402333**	**0.402333**

Table 6.4: Performance of Different Algorithms for Hepatitis Dataset using External Cluster Validity Indices

Different Algorithms	Precision	Sensitivity	Specificity	Accuracy	Jaccard Coefficient	Dice Coefficient	G Measure	Matthews Corr Coeff	Time (msec)
HCM	0.402685	0.491803	0.491803	**0.794702**	0.284361	0.442805	0.445019	-0.056486	11
FCM	0.522351	0.503430	0.503430	0.780645	0.344733	0.512716	0.512803	0.017511	151
PCM	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-
RCM	0.415435	0.421494	0.421494	0.632258	0.264577	0.418443	0.418454	-0.162958	38
RFCM	0.599714	0.579649	0.579649	0.754839	0.417948	0.589511	0.589596	0.178237	20
RPCM	0.631432	0.667429	0.667429	0.729032	0.480310	0.648932	0.649181	0.296685	39
RFPCM	0.666417	0.753938	0.753938	0.664516	0.547366	0.707481	0.708828	0.411143	45
KHCM	0.654882	0.749548	0.749548	0.774194	0.537309	0.699025	0.700618	0.393195	41
KFCM	0.655836	0.721164	0.721164	0.722581	0.523171	0.686950	0.687725	0.371296	107
KPCM	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-
IT2FCM	0.500000	0.500000	0.500000	0.500000	0.333333	0.500000	0.500000	0.000000	29
RHFCM	0.522351	0.503430	0.503430	0.780645	0.344733	0.512716	0.512803	0.017511	9
RH-IT2FCM	**0.745448**	**0.767149**	**0.767149**	**0.832258**	**0.607902**	**0.756143**	**0.756221**	**0.512138**	17

Table 6.5: Performance of Different Clustering Algorithms for Wine Dataset using Internal Cluster Validity Indices

Different Algorithms	Silhouette Index			DB Index			Dunn Index			Xie-Beni Index		
	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best
HCM	0.454865	0.459409	0.459984	1.094385	1.089495	1.089495	1.496322	1.607238	1.607238	0.555075	0.467187	0.467187
FCM	**0.468277**	0.468277	0.468277	1.713142	1.712765	1.712765	0.976302	0.976114	0.976114	0.856523	0.856694	0.856694
PCM	0.445623	0.456674	0.456674	1.754753	1.732530	1.732530	0.974532	0.987434	0.987434	0.857763	0.855854	0.855854
FPCM	0.398843	0.412643	0.412643	1.754753	1.740873	1.738784	0.987854	0.986532	0.986532	0.828312	0.817323	0.817323
RCM	**0.468277**	0.459409	0.459984	1.091912	1.089495	1.088743	1.496322	1.607238	1.607238	0.555075	0.467187	0.467092
RFCM	0.339662	0.351543	0.351543	1.457743	1.436341	1.436341	0.769076	0.864634	0.864634	0.464801	1.092105	1.092105
RPCM	0.460718	0.463201	0.463201	1.038543	0.953456	0.953456	1.661451	1.640642	1.640642	0.488476	0.399918	0.399918
RFPCM	**0.468277**	0.468277	0.468277	0.939983	0.938732	0.938732	1.765002	1.765413	1.765413	0.468815	0.469031	0.459843
KHCM	0.137163	0.290243	0.290243	1.984530	1.974163	1.974163	0.648843	0.654326	0.655604	0.557386	0.458998	0.458384
KFCM	0.423621	0.466425	0.466425	1.817232	1.744468	1.744468	0.940754	0.963262	0.963334	0.911652	0.870736	0.870263
KPCM	0.398832	0.436452	0.436452	1.760342	1.765632	1.765632	0.953327	0.961327	0.961327	0.876436	0.845320	0.845320
KFPCM	0.389832	0.412632	0.428874	1.760342	1.779432	1.728747	0.953327	0.967764	0.967764	0.865743	0.842743	0.842743
IT2FCM	0.398845	0.411721	0.411721	2.494534	2.493415	2.493415	1.487532	0.509090	0.509090	0.819127	1.047079	1.047079
RHFCM	0.466699	0.580923	0.580923	0.932277	0.820645	0.820645	1.781235	1.919670	1.919670	0.414516	0.399394	0.399394
RH-IT2FCM	0.466699	**0.580923**	**0.580923**	**0.930454**	**0.816985**	**0.816985**	**1.785007**	**1.930280**	**1.930280**	**0.412783**	**0.399076**	**0.399076**

Table 6.6: Performance of Different Algorithms for Wine Dataset using External Cluster Validity Indices

Different Algorithms	Precision	Sensitivity	Specificity	Accuracy	Jaccard Coefficient	Dice Coefficient	G Measure	Matthews Corr Coeff	Time (msec)
HCM	0.398338	0.652778	0.805082	0.726592	0.328694	0.494762	0.509928	0.396769	19
FCM	0.500234	0.651163	0.865992	0.829574	0.394511	0.565806	0.570731	0.483717	49
PCM	0.540686	0.553216	0.785784	0.737828	0.376348	0.546879	0.546915	0.349512	86
FPCM	0.794643	0.713615	0.831028	0.771536	0.602502	0.751952	0.753040	0.578971	58
RCM	0.431856	0.518103	0.766573	0.711610	0.308099	0.471064	0.473018	0.267309	18
RFCM	0.406005	0.581084	0.769586	0.689139	0.314076	0.478018	0.485719	0.300088	50
RPCM	0.835640	0.689928	0.836493	0.794038	0.607491	0.755825	0.759297	0.582521	54
RFPCM	0.858170	0.707317	0.866200	0.835616	0.633286	0.775475	0.779101	0.629268	66
KHCM	0.431856	0.518103	0.766573	0.711610	0.308099	0.471064	0.473018	0.267309	55
KFCM	0.540686	0.553216	0.785784	0.737828	0.376348	0.546879	0.546915	0.349512	125
KPCM	0.794643	0.713615	0.831028	0.771536	0.602502	0.751952	0.753040	0.578971	365
KFPCM	0.831562	0.679870	0.823259	0.774194	0.597577	0.748104	0.751900	0.558067	171
IT2FCM	0.454312	0.599667	0.776721	0.700375	0.348587	0.516966	0.521954	0.340166	197
RHFCM	0.398338	0.652778	0.805082	0.726592	0.328694	0.494762	0.509928	0.396769	30
RH-IT2FCM	**0.953396**	**0.957747**	**0.975027**	**0.966292**	**0.914915**	**0.955567**	**0.955569**	**0.929910**	23

Table 6.7: Performance of Different Clustering Algorithms for WBC Dataset using Internal Cluster Validity Indices

Different Algorithms	Silhouette Index			DB Index			Dunn Index			Xie-Beni Index		
	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best
HCM	0.428093	0.434212	0.434212	1.106560	1.086289	1.086289	1.255658	1.310884	1.310884	0.475945	0.468624	0.468624
FCM	0.425920	0.425920	0.425920	1.626298	1.626269	1.426634	0.848875	0.848852	0.848852	0.703446	0.703464	0.703464
PCM	-	-	-	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-	-	-	-
RCM	0.428093	0.434212	0.435086	1.106560	1.086289	1.086289	1.255658	1.310884	1.310884	0.475945	0.468624	0.468624
RFCM	0.421814	0.418504	0.418504	0.896027	2.277428	1.987354	0.605521	0.555836	0.615783	0.680453	0.650103	0.650103
RPCM	0.320247	0.322001	0.322897	1.372268	1.231676	1.231676	0.894517	0.998550	0.998550	0.794650	0.700551	0.700482
RFPCM	0.413709	0.414987	0.414987	1.012638	0.930141	0.930141	1.335316	1.452991	1.452991	0.445358	0.407416	0.407416
KHCM	0.427816	0.434212	0.434212	1.322564	1.119199	1.119199	1.049559	1.272286	1.272286	0.569580	0.482811	0.482811
KFCM	0.414035	0.423234	0.423234	5.072535	1.994911	1.994911	0.268875	0.690666	0.690666	2.247291	0.868424	0.868424
KPCM	-	-	-	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-	-	-	-
IT2FCM	0.426506	0.400999	0.400999	1.204492	0.827267	0.827267	1.261005	1.656762	1.656762	0.660249	0.350986	0.350453
RHFCM	0.428754	0.442349	0.442349	0.892735	0.824228	0.824228	1.556668	1.731043	1.731043	0.382632	0.346060	0.346060
RH-IT2FCM	**0.428754**	**0.442349**	**0.442349**	**0.888496**	**0.820615**	**0.820615**	**1.564108**	**1.738685**	**1.738685**	**0.380817**	**0.344545**	**0.344545**

Table 6.8: Performance of Different Algorithms for WBC Dataset using External Cluster Validity Indices

Different Algorithms	Precision	Sensitivity	Specificity	Accuracy	Jaccard Coefficient	Dice Coefficient	G Measure	Matthews Corr Coeff	Time (msec)
HCM	0.916869	0.894529	0.894529	0.912127	0.827420	0.905561	0.905630	0.811090	108
FCM	0.919231	0.876989	0.876989	0.903339	0.814245	0.897613	0.897862	0.795098	181
PCM	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-
RCM	0.737901	0.531176	0.531176	0.648506	0.446865	0.617701	0.626063	0.172241	74
RFCM	0.860606	0.674528	0.674528	0.757469	0.608091	0.756289	0.761907	0.501741	51
RPCM	0.806840	0.771220	0.771220	0.806678	0.651020	0.788628	0.788829	0.576962	206
RFPCM	0.908467	0.811321	0.811321	0.859402	0.750011	0.857150	0.858521	0.713202	105
KHCM	0.764825	0.754558	0.754558	0.778559	0.612457	0.759657	0.759674	0.519282	538
KFCM	0.791420	0.736728	0.736728	0.783831	0.616939	0.763095	0.763584	0.525308	484
KPCM	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-
IT2FCM	0.862555	0.762314	0.762314	0.843434	0.679743	0.809342	0.810887	0.616776	241
RHFCM	**0.919231**	0.876989	0.876989	0.903339	0.814245	0.897613	0.897862	0.795098	48
RH-IT2FCM	0.912882	**0.901677**	**0.901677**	**0.913884**	**0.830236**	**0.907245**	**0.907262**	**0.814482**	57

Table 6.9: Performance of Different Clustering Algorithms for Soybean-Small Dataset using Internal Cluster Validity Indices

Different Algorithms	Silhouette Index			DB Index			Dunn Index			Xie-Beni Index		
	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best
HCM	0.398941	0.411669	0.411669	2.991039	2.836326	2.836326	0.375823	0.409733	0.409733	2.176264	1.964814	1.964814
FCM	0.494149	0.592235	0.592235	2.984225	2.910796	2.910796	0.337131	0.333524	0.333524	2.042775	2.062835	2.062835
PCM	-	-	-	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-	-	-	-
RCM	0.398941	0.411669	0.411669	2.991039	2.836326	2.836326	0.375823	0.409733	0.409733	2.176264	1.964814	1.964814
RFCM	0.326949	0.407847	0.407847	1.773823	1.592647	1.592647	0.754647	0.822524	0.822524	0.859565	0.937256	0.937256
RPCM	0.426157	0.417025	0.417025	1.704412	1.365586	1.365586	0.512183	0.572513	0.598465	1.160176	1.095631	1.095631
RFPCM	0.337927	0.356390	0.423753	1.640623	1.431658	1.348745	0.645874	0.681326	0.781436	0.831247	0.970959	0.970959
KHCM	0.382465	0.411669	0.411669	2.567320	2.838017	2.838017	0.378865	0.409488	0.409488	1.803767	1.965995	1.965995
KFCM	0.426684	0.592235	0.592235	2.874753	3.264306	3.264306	0.386483	0.292310	0.345374	2.375191	2.330244	2.319832
KPCM	-	-	-	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-	-	-	-
IT2FCM	0.561812	0.604013	0.604013	**1.136483**	0.659587	0.659587	0.928342	2.014442	2.014442	0.722398	0.360893	0.360893
RHFCM	0.520026	0.604013	0.604013	1.388820	0.659620	0.659597	0.910360	2.014393	2.014393	0.669864	0.352306	0.352306
RH-IT2FCM	**0.561812**	**0.604013**	**0.604013**	1.193643	**0.659495**	**0.659495**	**0.985627**	**2.014674**	**2.014674**	**0.631787**	**0.352187**	**0.352187**

Table 6.10: Performance of Different Algorithms for Soybean-Small Dataset using External Cluster Validity Indices

Different Algorithms	Precision	Sensitivity	Specificity	Accuracy	Jaccard Coefficient	Dice Coefficient	G Measure	Matthews Corr Coeff	Time (msec)
HCM	0.644444	0.569444	0.950466	0.826389	0.433309	0.604627	0.605784	0.525660	6
FCM	0.647616	0.585142	**0.984030**	0.835902	0.443831	0.614796	0.615587	0.536827	86
PCM	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-
RCM	0.697253	0.711765	0.902252	0.851064	0.543727	0.704434	0.704472	0.604684	26
RFCM	0.727778	0.711765	0.894369	0.851064	0.562112	0.719682	0.719727	0.613968	10
RPCM	0.688889	0.682353	0.891892	0.829787	0.521613	0.685605	0.685613	0.572539	22
RFPCM	0.747159	0.747059	0.902703	0.861702	0.596308	0.747109	0.747109	0.649362	20
KHCM	0.597222	0.647059	0.872523	0.819149	0.450476	0.621142	0.621641	0.497057	28
KFCM	0.606993	0.656484	0.871612	0.852637	0.460674	0.630769	0.631254	0.513135	135
KPCM	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-
IT2FCM	0.747159	0.747059	0.902703	0.861702	0.596308	0.747109	0.747109	0.649362	12
RHFCM	0.959422	0.915245	0.953064	0.945308	0.881137	0.936813	0.937073	0.937424	5
RH-IT2FCM	**0.973684**	**0.950000**	0.983333	**0.978723**	**0.926218**	**0.961696**	**0.961769**	**0.946256**	9

Table 6.11: Performance of Different Clustering Algorithms for Dermatology Dataset using Internal Cluster Validity Indices

Different Algorithms	Silhouette Index			DB Index			Dunn Index			Xie-Beni Index		
	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best
HCM	0.352390	0.425116	0.425116	2.267608	1.263412	1.263412	0.275098	0.536526	0.536526	0.848711	0.848712	0.848712
FCM	0.373236	0.425283	0.425283	2.167843	1.148327	1.148327	0.336523	0.628836	0.628836	0.829704	0.826304	0.826304
PCM	-	-	-	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-	-	-	-
RCM	0.352390	0.425116	0.425116	2.267608	1.263412	1.167854	0.275098	0.536526	0.536526	0.848711	0.848712	0.848712
RFCM	0.288958	0.421826	0.421826	2.380129	0.995591	0.995591	0.342474	0.817967	0.817967	1.154534	0.746652	0.746652
RPCM	0.244848	0.372084	0.398787	1.384498	0.978141	0.978141	0.383185	0.758270	0.758270	0.913533	0.708261	0.708261
RFPCM	0.353377	0.383645	0.426805	**1.195673**	0.974073	0.974073	0.411822	0.817063	0.817998	0.784675	0.732358	0.702373
KHCM	0.352390	0.425116	0.425116	2.280444	1.271467	1.271467	0.423732	0.534899	0.534899	0.854316	0.854061	0.847734
KFCM	0.358943	0.424384	0.424384	1.563056	1.015566	1.015566	0.454761	0.618342	0.618342	0.901563	0.840473	0.819845
KPCM	-	-	-	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-	-	-	-
IT2FCM	0.363659	0.388924	0.388924	1.600094	1.821962	1.620896	0.453312	0.396590	0.503353	0.471385	0.424351	0.424351
RHFCM	0.381278	0.461566	0.461566	1.608252	0.872532	0.872532	0.459365	1.016175	1.016175	0.264589	0.266338	0.266338
RH-IT2FCM	**0.381278**	**0.461566**	**0.461566**	1.606572	**0.870683**	**0.867109**	**0.459374**	**1.020987**	**1.020987**	**0.264486**	**0.266267**	**0.266266**

Table 6.12: Performance of Different Algorithms for Dermatology Dataset using External Cluster Validity Indices

Different Algorithms	Precision	Sensitivity	Specificity	Accuracy	Jaccard Coefficient	Dice Coefficient	G Measure	Matthews Corr Coeff	Time (msec)
HCM	0.845612	0.802436	0.911273	0.886097	0.699897	0.823458	0.823741	0.815354	77
FCM	0.867564	0.522998	0.935817	0.892532	0.484330	0.652591	0.673598	0.848323	764
PCM	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-
RCM	0.845612	0.802436	0.911273	0.886097	0.699897	0.823458	0.823741	0.815354	51
RFCM	0.900412	0.877325	0.975753	0.958106	0.799725	0.888719	0.888794	0.854875	131
RPCM	0.750779	0.802506	0.966832	0.944444	0.633695	0.775781	0.776212	0.746692	264
RFPCM	0.914181	0.909269	0.982632	0.969945	0.837759	0.911718	0.911722	0.889599	135
KHCM	0.894814	0.874742	0.975732	0.958106	0.793182	0.884664	0.884721	0.852320	356
KFCM	0.905539	0.886112	0.996138	0.966808	0.811135	0.895720	0.895773	0.861373	976
KPCM	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-
IT2FCM	0.750779	0.802506	0.966832	0.944444	0.633695	0.775781	0.776212	0.746692	165
RHFCM	0.905188	0.884070	0.977213	0.960784	0.809143	0.894504	0.894567	0.863726	56
RH-IT2FCM	**0.937545**	**0.949584**	**0.990542**	**0.983607**	**0.893090**	**0.943526**	**0.943545**	**0.933182**	140

Table 6.13: Performance of Different Clustering Algorithms for Echocardiogram Dataset using Internal Cluster Validity Indices

Different Algorithms	Silhouette Index			DB Index			Dunn Index			Xie-Beni Index		
	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best
HCM	0.338895	0.347458	0.347458	1.701545	1.656441	1.656441	0.957477	1.005150	1.005150	0.810862	0.783694	0.783694
FCM	0.332317	0.332317	0.332317	5.270000	5.270189	4.496060	0.303970	0.303959	0.458736	2.540030	2.540121	2.450626
PCM	-	-	-	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-	-	-	-
RCM	0.338895	0.347458	0.347458	1.701545	1.656441	1.656441	0.957477	1.005150	1.005150	0.810862	0.783694	0.758735
RFCM	0.355183	0.355183	0.355183	1.272431	1.017144	1.017144	1.340628	1.617517	1.617517	0.597484	0.470903	0.470903
RPCM	0.286535	0.286535	0.301275	1.418267	1.395213	1.395213	0.958506	0.974405	1.003332	0.818818	0.805488	0.795434
RFPCM	0.328127	0.328127	0.328127	1.172908	1.145259	1.145259	1.321577	1.351828	1.351828	0.564364	0.550943	0.550943
KHCM	0.346567	0.347458	0.347458	1.651965	1.659443	1.659443	0.988726	1.003332	1.003332	0.773612	0.785114	0.785114
KFCM	0.332317	0.332317	0.332317	5.357648	5.439008	4.585743	0.298951	0.294445	0.448787	2.582198	2.621354	2.450626
KPCM	-	-	-	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-	-	-	-
IT2FCM	0.303040	0.359450	0.359450	1.599087	1.346145	1.346145	1.403667	0.918457	0.918457	0.582473	0.466804	0.466804
RHFCM	0.333537	0.499902	0.499902	1.159412	0.641516	0.641516	1.391451	2.579945	2.579945	0.546821	0.383523	0.383523
RH-IT2FCM	**0.387536**	**0.499902**	**0.499902**	**1.148891**	**0.640001**	**0.640001**	**1.404319**	**2.587365**	**2.587365**	**0.541873**	**0.382433**	**0.382433**

Table 6.14: Performance of Different Algorithms for Echocardiogram Dataset using External Cluster Validity Indices

Different Algorithms	Precision	Sensitivity	Specificity	Accuracy	Jaccard Coefficient	Dice Coefficient	G Measure	Matthews Corr Coeff	Time (msec)
HCM	0.445833	0.494767	0.494767	0.552846	0.306359	0.469027	0.469663	-0.033674	28
FCM	0.521654	0.502582	0.502582	0.541985	0.344032	0.511940	0.512029	0.014955	51
PCM	-	-	-	-	-	-	-	-	-
FPCM	-	-	-	-	-	-	-	-	-
RCM	0.522201	0.522066	0.522066	0.526718	0.353302	0.522133	0.522133	0.044267	15
RFCM	0.536667	0.536150	0.536150	0.541985	0.366501	0.536408	0.536408	0.072815	43
RPCM	0.521654	0.502582	0.502582	0.541985	0.344032	0.511940	0.512029	0.014955	46
RFPCM	0.534810	0.533568	0.533568	0.541985	0.364431	0.534188	0.534189	0.068367	16
KHCM	0.541687	0.539319	0.539319	0.549618	0.370332	0.540500	0.540502	0.080972	15
KFCM	0.534033	0.534272	0.534272	0.534351	0.364398	0.534152	0.534152	0.068305	160
KPCM	-	-	-	-	-	-	-	-	-
KFPCM	-	-	-	-	-	-	-	-	-
IT2FCM	0.540060	0.536669	0.536669	0.534375	0.368325	0.538359	0.538362	0.126683	171
RHFCM	0.557215	0.550822	0.550822	0.564885	0.383126	0.554000	0.554009	0.107847	5
RH-IT2FCM	**0.566964**	**0.566667**	**0.566667**	**0.566667**	**0.395493**	**0.566815**	**0.566815**	**0.133631**	16

Table 6.15: Performance of Different Clustering Algorithms for Ecoli Dataset using Internal Cluster Validity Indices

Different Algorithms	Silhouette Index			DB Index			Dunn Index			Xie-Beni Index		
	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best	Random	Proposed	Best
HCM	0.208935	0.373644	0.373644	1.171650	1.130590	1.130590	0.042862	0.044073	0.044073	0.867759	0.778198	0.778198
FCM	0.177301	0.370416	0.370416	4.487675	3.533066	2.898320	0.037592	0.009899	0.025463	3.537829	1.776161	1.776161
PCM	0.216654	0.368974	0.368974	1.678767	1.465832	1.465832	0.050088	0.050084	0.050084	1.556475	0.798403	0.798403
FPCM	0.238763	0.370417	0.370417	1.590085	1.458703	1.458703	0.051286	0.054754	0.054754	1.456638	0.780054	0.780054
RCM	0.202967	0.373644	0.373644	1.171650	1.130590	1.130590	0.042867	0.044073	0.044073	0.867759	0.778198	0.778198
RFCM	0.202119	0.338057	0.338057	2.404863	1.244729	1.244729	0.042591	0.049627	0.049627	0.938144	0.706217	0.706217
RPCM	0.249263	0.295634	0.317623	2.589456	1.794745	1.794745	0.055934	0.053663	0.053663	0.936140	1.018251	0.983645
RFPCM	0.249595	0.384262	0.384262	2.369853	1.072606	1.072606	0.058566	0.053599	0.053599	0.845414	0.657893	0.657893
KHCM	0.233321	0.373799	0.373799	1.173415	1.112996	1.112996	0.040487	0.045870	0.045870	0.858396	0.778177	0.778177
KFCM	0.172692	0.369048	0.369048	3.159525	3.759786	2.984650	0.041983	0.009777	0.009777	1.991736	1.849966	1.249952
KPCM	0.230878	0.364221	0.364221	1.896643	1.697634	1.688730	0.051274	0.051274	0.051274	1.280045	0.648743	0.648743
KFPCM	0.245587	0.364221	0.364221	1.879045	1.657843	1.657843	0.051274	0.051676	0.051676	1.276584	0.638854	0.638854
IT2FCM	0.244358	0.369048	0.369048	1.020515	1.012757	1.012757	0.030487	0.031578	0.031578	0.707594	0.696566	0.696566
RHFCM	0.267245	0.361346	0.381923	0.901427	0.901427	0.901427	0.065800	0.080037	0.080157	0.467285	0.334219	0.334219
RH-IT2FCM	**0.268327**	**0.384262**	**0.384262**	**0.901005**	**0.900281**	**0.900281**	**0.066089**	**0.080409**	**0.080409**	**0.467309**	**0.333923**	**0.333923**

Table 6.16: Performance of Different Algorithms for Ecoli Dataset using External Cluster Validity Indices

Different Algorithms	Precision	Sensitivity	Specificity	Accuracy	Jaccard Coefficient	Dice Coefficient	G Measure	Matthews Corr Coeff	Time (msec)
HCM	0.583879	0.552142	0.940834	0.888393	0.396226	0.567567	0.567789	0.503739	128
FCM	0.583486	0.551384	0.931354	0.883867	0.395655	0.566981	0.567208	0.493991	447
PCM	0.489943	0.487248	0.865571	0.840030	0.323269	0.488592	0.488594	0.301919	324
FPCM	0.504060	0.505931	0.882660	0.878265	0.337787	0.504994	0.504995	0.302379	373
RCM	**0.583879**	0.552142	0.940834	0.888393	0.396226	0.567567	0.567789	0.503739	113
RFCM	0.423463	0.426598	0.931817	0.875000	0.269861	0.425025	0.425028	0.351534	112
RPCM	0.494906	0.551948	0.939157	0.884673	0.353064	0.521873	0.522649	0.440676	144
RFPCM	0.563048	0.559072	0.943139	0.894345	0.389905	0.561053	0.561056	0.499694	155
KHCM	0.547832	0.422637	0.917158	0.875744	0.313335	0.477159	0.481180	0.503212	215
KFCM	0.517380	0.399428	0.881699	0.854748	0.291002	0.450816	0.454594	0.488805	1185
KPCM	0.468936	0.453134	0.928673	0.872024	0.299461	0.460900	0.460967	0.387604	694
KFPCM	0.476397	0.460010	0.932104	0.888275	0.305534	0.468060	0.468132	0.388671	744
IT2FCM	0.468936	0.453134	0.928673	0.872024	0.299461	0.460900	0.460967	0.387604	926
RHFCM	0.571641	0.560508	0.943122	0.893601	0.394720	0.566020	0.566047	0.504840	56
RH-IT2FCM	0.573088	**0.566677**	**0.945749**	**0.897321**	**0.398468**	**0.569864**	**0.569873**	**0.512045**	118

kernel-based c-means clustering algorithms, with respect to both internal and external cluster validity indices used.

2. The proposed RH-IT2FCM algorithm achieves better results with lesser or comparable time than that obtained using interval type-2 fuzzy c-means, irrespective of the data sets and quantitative indices used. The hypercuboid approach, employed in RH-IT2FCM algorithm, helps to partition the objects in lower approximation and boundary region without using any user specified parameter, giving better results in terms of clustering.

3. It is also seen that the RH-IT2FCM algorithm attains better results with lesser time than that obtained using rough hypercuboid based fuzzy c-means, on eight data sets with respect to both internal and external cluster validity indices used. Inclusion of IT2 fuzzy approach in RH-IT2FCM helps to manage the uncertainty present in the fuzzy membership function.

4. Some of the existing algorithms like PCM, FPCM, KPCM, and KFPCM generate coincident clusters in most of the cases. In Tables 6.1 - 6.16, they are marked with '-'.

The best performance of the proposed RH-IT2FCM over other clustering algorithms, in terms of both internal and external cluster validity indices, is achieved because (i) the rough hypercuboid approach, used in the RH-IT2FCM, computes the lower approximation and boundary region implicitly for the clusters without using any threshold value; (ii) the concept of crisp lower approximation and interval type-2 fuzzy boundary of the RH-IT2FCM algorithm deals with uncertainty, incompleteness, and vagueness in cluster definition; and (iii) the IT2 fuzzy approach helps to manage the uncertainty present in the fuzzy membership function.

6.5.6 Execution Time

Moreover, Tables 6.2, 6.4, 6.6, 6.8, 6.10, 6.12, 6.14 and 6.16 provide the execution time of different algorithms on eight data sets. The results reported in Tables 6.2, 6.4, 6.6, 6.8, 6.10, 6.12, 6.14 and 6.16 establish the fact that the proposed algorithm provides good clustering results in lower or comparable time than most of the existing methods, irrespective of the data sets used. The lower execution time of the proposed algorithm is achieved due to its low computational complexity with the use of rough hypercuboid approach.

6.6 Conclusion

The contribution of the chapter is three-fold, namely,

1. the development of a novel clustering algorithm, integrating judiciously the c-means algorithm, rough hypercuboid approach, probabilistic memberships of fuzzy sets, and IT2 fuzzy approach, to overcome the problems of uncertainty present in data as well as in determining the fuzzy membership function;

2. formulation of a new cluster prototype initialization method in order to circumvent efficiently the initialization of cluster prototypes and local minima problem of iterative refinement clustering like c-means; and

3. demonstrating the effectiveness of the proposed method, along with a comparison with existing c-means algorithms, on several real life data sets using some well known cluster validity indices.

The proposed algorithm is formulated combining the utility of both rough sets and IT2 fuzzy sets for knowledge discovery tasks. The proposed prototype initialization method achieves significantly better result in 84.90% cases than random initialization. The proposed initialization method also provides the best performance in 80.47% cases for optimum parameter values. Moreover, the proposed clustering algorithm outperforms other c-means algorithms, irrespective of the data sets and quantitative indices used. The proposed clustering performs better than any other c-means algorithms in 93.75% and 91.19% cases, respectively, with respect to internal and external cluster validity indices, and generates relevant clusters in lesser or comparable time.

The next chapter presents the application of clustering in one of the important areas of bioinformatics, which is grouping functionally similar miRNAs. Similar to other clustering problems, in miRNA clustering, one of the important issues is which time points (attributes) of the data should be used for grouping the miRNAs. In this regard, the clustering algorithm proposed in the current chapter is combined with the feature selection algorithm proposed in Chapter 4 to develop a simultaneous feature selection and clustering framework, which is presented in the next chapter.

Chapter 7

Grouping Functionally Similar microRNAs from Microarray Data

7.1 Introduction

MicroRNAs (miRNAs) are a recently discovered class of small single-stranded non-coding RNA molecules, which play an important regulatory role in the gene expression of animals and plants. Various studies have revealed that miRNAs tend to cluster on chromosomes. The members of a cluster that are in close proximity on chromosomes are highly likely to be processed as co-transcribed units. Therefore, a large proportion of miRNAs are co-expressed [70]. Expression profiling of miRNAs generates a huge volume of data. Complicated gene interaction networks increase the challenges of comprehending and interpreting the resulting mass of data, which often consists of millions of measurements.

The genes of miRNAs are often organized into clusters in the genome. It has been reported that at a very conservative maximum inter-miRNA distance of 1 kb, over 30% of all miRNAs are organized into clusters. Expression analyses showed strong positive correlations among the closely located miRNAs, indicating that they may be controlled by common regulatory element(s). In fact, experimental evidence demonstrated that clustered miRNAs form an operon-like gene structure and they are transcribed from a common promoter. Existence of co-expressed miRNAs is also demonstrated using expression profiling analysis. Expression data of miRNAs can be used to detect clusters of miRNAs as it is suggested that co-expressed miRNAs are co-transcribed, so they should have a similar expression pattern.

A miRNA expression data set can be represented by an expression table, where each row corresponds to one particular miRNA, each column to a sample or a time point, and each entry of the matrix is the measured expression level of a particular miRNA in a sample or a time point, respectively [70, 108]. The complex networks of miRNA-

mRNA interaction greatly increase the challenges of comprehending and interpreting the resulting mass of data. A first step towards addressing this challenge is the use of clustering techniques, which is essential in the pattern recognition process to reveal natural structures and identify interesting patterns in the underlying data [226].

Clustering, as explained in Chapter 6, has retained high interest for analyzing miRNA data, where the motivation is to group together the co-functional and co-regulated miRNAs into clusters according to their expression profiles [329, 405, 408]. The grouping is done in such a way that the degree of association between two miRNAs from the same cluster is maximum and minimum for the miRNAs from different clusters [108]. To understand the role of miRNAs in different cellular processes and diseases, and the mechanism of repression of mRNA or gene translation, clustering techniques have proven to be helpful. This approach may further be used for understanding the functions of many miRNAs for which information has not been previously available [117, 491]. Furthermore, co-expressed miRNAs in the same cluster are likely to be involved in the same cellular processes, and a strong correlation of expression patterns between those miRNAs indicates co-regulation.

Each miRNA consists of large number of time points. The most important time points can be used to group the miRNAs, whereas the other time points may be redundant or irrelevant. In most of the miRNA data sets, a significant number of redundant and insignificant time points are present, which must be removed. The number of subset combinations for selecting m time points from a set of N time points is $N!/[m!(N-m)!]$, which increases rapidly with the growth of N, increasing the processing time for the feature selection algorithms while selecting the optimal set of time points. Hence, feature selection, which has also been explained in Chapters 3 and 4, can be used for selecting non-redundant and relevant features (time points) before clustering the miRNAs. Here, the main objective is to retain the optimum salient characteristics necessary for the grouping of co-expressed miRNAs and to reduce the time points so that effective and easily computable algorithms can be devised for efficient clustering [101, 112, 382].

Supervised feature selection algorithms [314,326,327] can be used for the data sets that consist of decision attribute or class label. The conventional rough set theory and theory of rough-fuzzy computing have been successfully applied to supervised feature selection of real valued data set [76, 199, 220, 223, 321, 323, 500], which has already been discussed in Chapters 3 and 4. Exhaustive studies have also been done on unsupervised feature selection for data sets, where no decision attribute is present. Unsupervised feature selection using feature similarity (FSFS) [354] and unsupervised feature selection for multi-cluster data (MCFS) [66] are two mostly used algorithms falling under this class.

Like all other unsupervised cases, selection of time points for grouping the miRNAs is also difficult, because, unlike in supervised learning, there is no class label for the miRNA data, and thus no obvious criterion to guide the search. For supervised feature selection,

class labels are important. Similarly, for clustering, the most relevant features are requited. So, feature selection and clustering are inter-dependent. The performance may be significantly improved if the feature selection and clustering task can be done simultaneously, rather than sequentially. For this purpose, expectation-maximization based techniques and some other methods can be found in literature [119, 271, 486] with high complexities.

In this chapter, the algorithm proposed in Section 6.3.5 of Chapter 6 is extended and applied for discovering co-expressed miRNAs from huge miRNA expression data set, where the algorithm simultaneously selects the most important time points while grouping the miRNAs. The extended version is termed as interval type-2 fuzzy-rough hypercuboid based simultaneous feature selection and clustering (IT2FRH-SFSCL), which employs rough hypercuboid approach to provide a means by which real valued noisy data can be effectively reduced without the need of any user-specified information. Using the concept of hypercuboid equivalence partition matrix, reported in Section 6.2 of Chapter 6, the lower approximation and boundary region are computed corresponding to the attributes, and the features thus selected can be used for clustering. Hence, the only information required in the proposed method is in the form of equivalence classes for each attribute, which can be automatically derived from the data set. The IT2 fuzzy-rough set, reported in Section 4.2 of Chapter 4, is used for the boundary region to model the uncertainty present in the fuzzy membership function. Several quantitative measures are used to evaluate the performance of the proposed extension. The optimum values of different parameters are estimated using the procedure described in Section 6.5.1 of Chapter 6. The effectiveness of the proposed algorithm, along with a comparison with other algorithms, is demonstrated on some miRNA data sets using some standard cluster validity indices. Moreover, the GO Term Finder [60] is used to determine biological significance of generated clusters. Some of the results, presented in this chapter, are also reported in [146, 147].

The structure of the rest of this chapter is as follows: In Section 7.2, the proposed algorithm, termed as IT2FRH-SFSCL, is presented, combining the merits of IT2FR-MRMS of Chapter 4 and RH-IT2FCM of Chapter 6. In Section 7.3, the performance of the proposed algorithm is demonstrated on several miRNA data, along with a few case studies and a comparison with other clustering algorithms. Concluding remarks are given in Section 7.4.

7.2 Simultaneous Feature Selection and Clustering

One of the most frequently used clustering techniques is objective function based clustering, where the distance between a pattern and corresponding cluster prototype needs to be minimized iteratively. The calculation of distances depends on the selection of attributes. In case of traditional c-means algorithm, all the features are given the same weighage, so that they can contribute equally for grouping the samples. But, all the attributes

are not equally important to generate an optimum grouping, which is more closer to the natural clusters present in the data set. There may be some insignificant and irrelevant features present in the data set, which may prevent the clustering algorithm to produce the optimum partitions. To get good clustering results, the aforementioned features must be removed while calculating the distances, or given a less priority. Hence, the features may be ranked according to their importance and the most important features may be used to group the data set. Before clustering, feature selection is very difficult to perform in most of the cases. Unsupervised feature selection algorithms can only be used at this step, which select the features using the inherent properties, and does not require any class information. But, the performance of the unsupervised feature selection approaches are not as good as the supervised counterparts in general, as no class label is used which can guide the searching process. Supervised feature selection requires class labels, and clustering requires the optimum features. So, they are inter-dependent and need to be performed simultaneously, rather than sequentially.

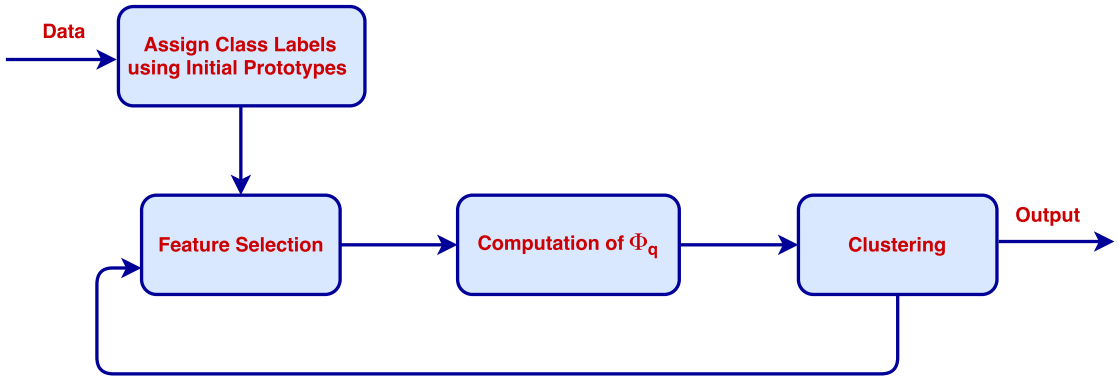


Figure 7.1: Outline of the proposed approach

As mentioned in Chapter 4, IT2 based fuzzy-rough maximum-relevance maximum-significance (IT2FR-MRMS) is a promising technique for attribute selection, which is a supervised method, where both relevance and significance are maximized. So, this method is merged judiciously with the clustering method (RH-IT2FCM) proposed in Chapter 6 to construct an iterative algorithm, that can simultaneously select the optimum set of attributes necessary for clustering while clustering the samples in the data set. The proposed extension towards simultaneous feature selection and clustering is outlined in Figure 7.1.

Feature Selection Step: The method proposed in Section 4.3 of Chapter 4 is used here for attribute selection. The method uses IT2 fuzzy-rough sets and maximum-relevance maximum-significance criterion for attribute selection. At each iteration, one feature $\mathcal{A}_q$

is selected that maximizes the following condition:

$$\Phi_q = \omega\gamma_{\mathcal{A}_q}(\mathbb{D}) + \frac{(1-\omega)}{|\mathbb{S}|}\sum_{\mathcal{A}_p\in\mathbb{S}}\sigma_{\{\mathcal{A}_p,\mathcal{A}_q\}}(\mathbb{D},\mathcal{A}_q). \tag{7.1}$$

where $\mathcal{A}_p \in \mathbb{S}$ are the features already selected in the previous iteration of feature selection. Lower and upper relevance and significance are used giving rise to two different equations (4.33) and (4.35) of Chapter 4. The feature selection process is iterated until the required number of features are selected. The values Φ_q acts as feature weight corresponding to feature $\mathcal{A}_q$, and is used in the next iteration to find out the distances from the centroid.

Clustering Step: The proposed technique uses the feature selection method discussed in Section 4.3 of Chapter 4 to form an iterative algorithm for simultaneously selecting the optimum features while grouping the objects. The rough-hypercuboid based interval type-2 fuzzy c-means method, proposed in Section 6.3 of Chapter 6, is used here for clustering.

Let $d_{ij} = ||x_j - v_i||$ be the distance between jth object x_j from the ith cluster center v_i. To integrate the feature selection with clustering, the feature weights (7.1) play an important role. Unlike the RH-IT2FCM, where same weights have been assigned to each feature, here, due to the simultaneous nature of feature selection and clustering, features are given different weightage according to their importance. These weights are used at the time of distance calculation. So d_{ij} can be represented as

$$d_{ij} = \sqrt{\sum \Phi_q(x_j - v_i)^2}, \tag{7.2}$$

where Φ_q represents the feature weights (7.1), calculated in the feature selection phase.

IT2FRH-SFSCL Method: The proposed algorithm starts by randomly choosing c objects as initial centroids of c clusters, and assigns each object x_j to the group that has the closest centroid. The values of the fuzzifiers $\acute{m}_1$ and $\acute{m}_2$ are selected. Objects are put into the class for which they have the closest centroid. The most relevant and significant features are selected using the IT2FR-MRMS algorithm, proposed in Section 4.3.3 of Chapter 4. The values from (4.33) and (4.35) of Chapter 4 are assigned to Φ_q, which is considered as feature weight in the next iteration to calculate the distances of the samples from the centroid. The objects are grouped using the RH-IT2FCM algorithm, proposed in Section 6.3.5 of Chapter 6, with the help of the features selected using IT2FR-MRMS algorithm. The above steps are repeated until no more change is observed for the centroids.

The main steps of the IT2FRH-SFSCL proceed as follows:

1. Set iteration counter $t = 1$. Assign initial prototypes $v_i^{(t)}$, $i = 1, 2, \cdots, c$. Choose the values for fuzzifiers $\acute{m}_1$ and $\acute{m}_2$. Set $\varepsilon > 0$ to a very small value. Set same weights

for all the features.

2. Assign each object x_j to the group that has the closest centroid. These labels are considered as class labels for Step 3.

3. Use the IT2FR-MRMS algorithm, proposed in Section 4.3.3 of Chapter 4, to select most relevant and significant features. Assign the values from (4.33) and (4.35) of Chapter 4 to Φ_q. The value of Φ_q is used as the feature weight (7.2) in Step 2 of the next iteration to find out the distances of samples from the centroid.

4. Use the RH-IT2FCM algorithm, proposed in Section 6.3.5 of Chapter 6, to group the objects using the features selected in Step 3.

5. Repeat Steps 2 to 4, by incrementing t, until no more new assignment can be made, that is, $\|v^{(t+1)} - v^{(t)}\| < \varepsilon$.

6. Stop.

For a data set with n objects and m attributes, the proposed technique has $\mathcal{O}(t(mnd\hat{c}^2\tilde{c}+cmn))$ time complexity, while the space complexity is $\mathcal{O}((n+c)m)$. Here $\hat{c}$ and $\tilde{c}$, respectively, denote the number of IT2 fuzzy equivalent classes corresponding to the condition attributes and decision attribute, and c denotes the number of clusters. d represents the number of selected features.

7.3 Experimental Results and Discussion

The performance of the proposed IT2FRH-SFSCL algorithm is compared extensively with that of different existing clustering algorithms and feature selection-clustering algorithms. In the later version, feature selection techniques are applied first on the data sets to select the most important features, and then clustering techniques are used to partition the data sets. As no class information is present before clustering, only unsupervised feature selection algorithms can be used for this purpose.

To compare the clustering performance, the proposed algorithm is compared with that of hard c-means (HCM) [216, 341], fuzzy c-means (FCM) [48], rough-fuzzy c-means (RFCM) [319], rough-hypercuboid based FCM (RHFCM) [146], and rough-hypercuboid based interval type-2 FCM (RH-IT2FCM). Unsupervised feature selection using feature similarity (FSFS) [354] (three different versions) and unsupervised feature selection for multi-cluster data (MCFS) [66] are used as the feature selection algorithms. Three versions of FSFS are denoted as FSFS1, FSFS2, and FSFS3, where the similarities between features are computed using correlation coefficient, linear regression error, and maximal information compression index, respectively. The proposed algorithm is also compared

with expectation-maximization based simultaneous feature selection and clustering (EM-SFSCL) [271] algorithm. All the algorithms are implemented in C language and executed in LINUX environment in a machine with Pentium Core i5, 2.4 GHz, 3 MB 12-way SA L3 cache, and 4 GB DDR3 RAM. The values of fuzzifiers $\acute{m}_1$ and $\acute{m}_2$ are varied taking the values from the set $\{1.1, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5\}$, for all values of $\acute{m}_2 \geq \acute{m}_1$. The value of weight parameter w for rough-fuzzy clustering is taken as 0.99 as suggested in [321].

The experimental results on five publicly available miRNA data sets are presented in this section. Subsequent discussions analyze the results with respect to four internal cluster validity indices, namely, Silhouette index [446], DB index [50], Dunn index [50], Xie-Beni index [539], and execution time. The quantitative indices are discussed in Appendix A.

7.3.1 Description of Data Sets

In this chapter, five publicly available miRNA expression data sets are used to compare the performance of different clustering methods. This subsection gives a brief description of the following five miRNA expression data sets, which are downloaded from Gene Expression Omnibus (www.ncbi.nlm.nih.gov/geo/).

GSE59973

Three pair of surgically removed human ESCC tissues and matched normal esophageal epithelial tissues (5cm away from tumor) were collected to make an Agilent microarray. All patients have no radiotherapy or chemotherapy before surgery. None of these three patients had distant metastasis. This data set contains 299 miRNAs and 6 time points.

GSE62018

Although formalin fixed paraffin embedded (FFPE) tissue is a major biological source in cancer research, it is challenging to work with due to macromolecular fragmentation and nucleic acid crosslinking. Therefore, it is important to characterise the quality of data that can be obtained from FFPE samples. This is a cross-comparison of technologies for the detection of miRNAs from FFPE samples of hepatoblastoma patients. This data set [74] contains 1733 miRNAs and 6 time points.

GSE70748

The aim of this study [442] was to identify differently expressed genes between C3 or C3156-181-peptide treated Isolated primary rat neonatal Schwann Cells. To elucidate the unresolved mechanism behind the promoting effect of C3156-181 on PNR, primary rat

neonatal SCs are cultured and treated for up to 72 h with C3 or C3156-181. This data set contains 1205 miRNAs and 32 time points.

GSE71302

The work [522] was done on expression profiles of miRNAs in clear cell renal cell carcinomas (ccRCCs) and in matched normal kidney tissues (NCTs) by using a miRNAs microarray platform which covers total 851 human miRNAs. This data set contains 10 time points.

GSE71107

The study [279] generated an artificially-designed interfering long non-coding RNA (lncRNAi). The adenovirus-expressed lncRNAi with high level in hepatocellular carcinoma (HCC) cells competes with OncomiR target genes to bind to and consume OncomiRs, thereby achieving the targeted anti-HCC efficacy. This data set contains 2549 miRNAs and 15 time points.

7.3.2 Optimum Values of Different Parameters

The parameters $\acute{m}_1$ and $\acute{m}_2$ ($\acute{m}_2 \geq \acute{m}_1$) of IT2FRH-SFSCL control the shape of both of the membership functions, giving the lower membership, $\underline{\mu}_{ij}$ and upper membership, $\overline{\mu}_{ij}$. The membership functions can be altered, by varying $\acute{m}_1$ and $\acute{m}_2$, keeping the values of other parameters fixed. The thresholds λ_1 and λ_2 in (6.38) and (6.40) of Chapter 6 also have a significant role for the selection of initial cluster prototypes for different clusters. They control the dissimilarity between the initial prototypes. Hence, they have direct influences on the performance of proposed technique as well as other c-means algorithms. The Davies-Bouldin (DB) index [446], described in Section 6.5.1 of Chapter 6, is used to find out the optimum value of the two fuzzy parameters, namely, $\acute{m}_1$ and $\acute{m}_2$, and two thresholds, namely, λ_1 and λ_2, for different data sets.

The optimum values of $\acute{m}_1^\star$, $\acute{m}_2^\star$, $\lambda_1^\star$ and $\lambda_2^\star$, obtained using (6.42) of Chapter 6, are {1.5, 2.5, 0.92, 0.90}, {1.5, 2.0, 0.94, 0.88}, {1.5, 2.0, 0.96, 0.90}, {1.5, 2.0, 0.90, 0.92}, and {1.1, 2.0, 0.94, 0.86}, respectively, for GSE59973, GSE62018, GSE70748, GSE71302, and GSE71107 data sets. In this regard, it should be noticed that the proposed IT2FRH-SFSCL technique delivers best performance at $\acute{m}_1 = \acute{m}_1^\star$, $\acute{m}_2 = \acute{m}_2^\star$, $\lambda_1 = \lambda_1^\star$, $\lambda_2 = \lambda_2^\star$ in 18 cases, out of 20, with respect to different cluster validity indices. However, the Dunn index provides the maximum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.92$, $\lambda_2 = 0.90$ for GSE62018 data and Xie-Beni index provides the minimum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.92$, $\lambda_2 = 0.86$ for GSE71107 data. Also, all other clustering algorithms achieve their best performance at $\acute{m}_1 = \acute{m}_1^\star$, $\acute{m}_2 = \acute{m}_2^\star$, $\lambda_1 = \lambda_1^\star$ and $\lambda_2 = \lambda_2^\star$ in 455 cases, out of 520 cases, irrespective of the cluster validity indices used. Hence, all the results reported

Table 7.1: Performance of Different Clustering Algorithms on GSE59973 Data Set

Different Algorithms		Silhouette Index	DB Index	Dunn Index	Xie-Beni Index	Time (msec)
HCM		0.289742	0.661235	1.105413	0.568851	423
FCM		0.315388	0.903619	0.869784	0.294521	1509
RFCM		0.220592	0.944311	1.048439	0.551269	449
RHFCM		0.343131	0.364713	3.228213	0.198967	15
RH-IT2FCM		0.467654	0.273430	3.243087	0.144281	46
HCM	FSFS1	0.328396	0.718340	0.987735	0.219335	134
	FSFS2	0.326388	1.004748	0.857738	0.157754	120
	FSFS3	0.460898	0.694666	1.490384	**0.130345**	112
	MCFS	0.464197	0.712584	1.434358	0.133528	73
FCM	FSFS1	0.317649	1.138140	0.551999	0.242396	208
	FSFS2	0.322354	1.002353	0.759853	0.162742	235
	FSFS3	0.444630	0.924469	0.964926	0.149832	281
	MCFS	0.369379	0.982925	0.811608	0.179696	408
RFCM	FSFS1	0.303325	0.875035	1.276236	0.172749	121
	FSFS2	0.223443	0.946677	1.147333	0.160321	181
	FSFS3	0.332146	0.675629	1.363763	0.147123	103
	MCFS	0.371426	0.738177	1.176380	0.153610	113
RHFCM	FSFS1	0.353201	1.474873	0.521303	1.669361	203
	FSFS2	0.402369	0.814362	0.896210	0.864319	335
	FSFS3	0.428507	0.806233	0.968426	0.596266	91
	MCFS	0.419968	0.772654	1.128582	0.635009	131
RH-IT2FCM	FSFS1	0.353201	1.474873	0.521303	1.669361	333
	FSFS2	0.396866	1.028285	0.734336	1.058635	308
	FSFS3	0.460513	0.800331	0.959169	0.824234	1103
	MCFS	0.421999	0.752471	1.272190	0.587574	1112
EM-SFSCL		0.618124	0.381241	1.813534	0.373643	4365
IT2FRH-SFSCL		**0.704496**	**0.261772**	**3.475758**	0.149452	2620

above establish the importance of proposed parameter optimization technique, irrespective of the clustering algorithms compared.

7.3.3 Performance Analysis

The performance of the proposed IT2FRH-SFSCL algorithm is compared with that of some existing c-means algorithms on five data sets with respect to several cluster validity indices. Tables 7.1 – 7.5 present the comparative assessment of these five clustering algorithms and four unsupervised feature selection algorithms, in terms of Silhouette, DB, Dunn, and Xie-Beni index. All the results correspond to optimum values of λ_1 and λ_2. The bold values in these tables signify the best values. Out of 100 cases, in 98 cases, the IT2FRH-SFSCL algorithm with proposed initialization method provides better results than other clustering algorithms, in terms of different cluster validity indices.

Tables 7.1 - 7.5 provide the comparative results of different feature selection and clustering algorithms with the proposed centroid initialization method discussed in Section 6.3.6 of Chapter 6 for five data sets. The results are analyzed to prove the performance improve-

Table 7.2: Performance of Different Clustering Algorithms on GSE62018 Data Set

Different Algorithms		Silhouette Index	DB Index	Dunn Index	Xie-Beni Index	Time (msec)
HCM		0.353208	0.519036	0.870872	0.162856	1684
FCM		0.330860	0.604450	0.240201	0.537453	5513
RFCM		0.390754	0.328626	1.534757	0.062063	497
RHFCM		0.370803	0.417677	1.799391	0.142839	64
RH-IT2FCM		0.701779	0.147534	**4.491506**	0.036446	694
HCM	FSFS1	0.416791	0.642356	0.672047	0.293471	480
	FSFS2	0.480483	0.672238	0.786330	0.283458	481
	FSFS3	0.480483	0.672238	0.786330	0.283458	519
	MCFS	0.436705	0.693082	0.740501	0.395858	498
FCM	FSFS1	0.405659	0.679384	0.290694	0.638365	2231
	FSFS2	0.455379	0.614785	0.239533	0.539183	2443
	FSFS3	0.455379	0.614785	0.239533	0.539183	2230
	MCFS	0.375043	0.738486	0.136885	1.526320	2491
RFCM	FSFS1	0.480879	0.728299	0.845717	0.060225	299
	FSFS2	0.546402	0.354178	2.627715	0.092150	238
	FSFS3	0.546402	0.354178	2.627715	0.092150	212
	MCFS	0.474536	0.688177	0.641210	0.381111	703
RHFCM	FSFS1	0.459512	0.785992	0.852253	0.094692	221
	FSFS2	0.531529	0.408949	2.333125	0.077365	315
	FSFS3	0.531529	0.408949	2.333125	0.077365	524
	MCFS	0.492046	0.584992	0.929862	0.168849	1034
RH-IT2FCM	FSFS1	0.459512	0.785992	0.852253	0.094692	351
	FSFS2	0.531529	0.408949	2.333125	0.077365	445
	FSFS3	0.531529	0.408949	2.333125	0.077365	559
	MCFS	0.492046	0.584992	0.929862	0.168849	1778
EM-SFSCL		0.776363	0.313643	1.857645	0.033896	2664
IT2FRH-SFSCL		**0.899141**	**0.127181**	4.317694	**0.032749**	1811

ment due to simultaneous feature selection and clustering over the traditional approach where those were done serially. Out of 80 cases each for GSE59973, GSE62018, GSE70748, GSE71302, and GSE71107 data sets, in 77, 80, 76, 80, and 80 cases, respectively, the proposed extension performs better in lesser or comparable time when compared with the other feature selection-clustering combinations, where they are performed in serial fashion. The best performance of the IT2FRH-SFSCL over the other serial algorithm combinations, in terms of the various cluster validity indices, is achieved because the simultaneous nature of feature selection and clustering helps to find out the best features necessary to group the objects in a data set while actually performing the grouping, which again proves the interdependency of feature selection and clustering in case of grouping the objects in unsupervised data sets.

The proposed IT2FRH-SFSCL method is finally compared with the existing expectation-maximization based simultaneous feature selection and clustering (EM-SFSCL) algorithm [271]. Tables 7.1 – 7.5 present the comparative results. Out of 20 cases, in 19 cases, the IT2FRH-SFSCL algorithm provides better results than other clustering algorithms and feature selection-clustering combinations, in terms of different cluster validity indices.

Table 7.3: Performance of Different Clustering Algorithms on GSE70748 Data Set

Different Algorithms		Silhouette Index	DB Index	Dunn Index	Xie-Beni Index	Time (msec)
HCM		0.361388	0.439571	0.163743	0.045934	1475
FCM		0.340258	0.476173	0.116263	0.061436	4575
RFCM		0.386205	0.424350	0.399105	0.051663	2897
RHFCM		0.486766	0.309504	0.236006	0.017004	398
RH-IT2FCM		0.683150	0.112586	0.536456	**0.003152**	1137
HCM	FSFS1	0.389045	0.593088	0.066897	0.056426	562
	FSFS2	0.459295	0.496253	0.126860	0.052137	599
	FSFS3	0.474050	0.503817	0.092048	0.049776	456
	MCFS	0.508604	0.447797	0.121478	0.044902	721
FCM	FSFS1	0.344417	0.641482	0.058740	0.065197	1382
	FSFS2	0.436407	0.548792	0.096269	0.061685	1783
	FSFS3	0.447135	0.517417	0.103628	0.064064	1232
	MCFS	0.470134	0.480161	0.084644	0.050848	3494
RFCM	FSFS1	0.388507	0.589641	0.092906	0.047353	2674
	FSFS2	0.500414	0.454022	0.240684	0.052528	1284
	FSFS3	0.497663	0.465570	0.202777	0.056274	1894
	MCFS	0.549590	0.443881	0.388724	0.060352	2010
RHFCM	FSFS1	0.374623	0.612204	0.050399	0.023086	483
	FSFS2	0.449901	0.567979	0.057486	0.017043	784
	FSFS3	0.505621	0.466573	0.474752	0.232932	658
	MCFS	0.629965	0.329080	0.219441	0.015501	733
RH-IT2FCM	FSFS1	0.374623	0.612204	0.050399	0.023086	1231
	FSFS2	0.449901	0.567979	0.057486	0.017043	983
	FSFS3	0.505621	0.466573	0.474752	0.232932	1327
	MCFS	0.629965	0.329080	0.219441	0.015501	2546
EM-SFSCL		0.767764	0.234231	0.412312	0.004345	2541
IT2FRH-SFSCL		**0.911553**	**0.088005**	**0.022504**	0.022504	1487

From all the results reported in Tables 7.1 - 7.5, the following conclusions can be drawn:

1. The proposed IT2FRH-SFSCL is better than any other c-means algorithms, with respect to different cluster validity indices used.

2. The IT2FRH-SFSCL algorithm achieves better results with lesser or comparable time than that obtained using feature selection-clustering combinations, irrespective of the data sets and quantitative indices used. The simultaneous nature of feature selection and clustering finds the best features from a data set while actually performing the grouping, giving better results in terms of clustering.

3. It is also seen that the proposed algorithm achieves better results with lesser or comparable time than that obtained using the expectation-maximization based simultaneous feature selection and clustering (EM-SFSCL) algorithm [271], irrespective of the data sets and quantitative indices used. The hypercuboid approach, employed in IT2FRH-SFSCL, helps to partition the objects in lower approximation and boundary region without using any user specified parameter, and interval type-2

Table 7.4: Performance of Different Clustering Algorithms on GSE71302 Data Set

Different Algorithms		Silhouette Index	DB Index	Dunn Index	Xie-Beni Index	Time (msec)
HCM		0.444104	0.537958	1.351018	0.258234	377
FCM		0.243719	1.428866	0.291935	0.349610	1953
RFCM		0.415036	0.431446	2.005782	0.106044	330
RHFCM		0.480111	0.374687	2.693452	0.165194	537
RH-IT2FCM		0.629885	0.198627	4.141907	0.049235	1885
HCM	FSFS1	0.286111	1.204049	0.351354	0.091463	412
	FSFS2	0.425945	0.868731	0.929613	0.523299	423
	FSFS3	0.504598	0.754178	0.907936	0.100768	354
	MCFS	0.539352	0.584627	1.575909	0.100987	358
FCM	FSFS1	0.265242	1.483048	0.478851	0.630535	2698
	FSFS2	0.254133	1.617545	0.422674	0.949326	2012
	FSFS3	0.307011	1.350037	0.737374	0.362283	1763
	MCFS	0.311798	1.171188	0.505643	0.636734	2031
RFCM	FSFS1	0.402949	0.770198	0.691932	0.105462	298
	FSFS2	0.506465	0.543916	0.923705	0.115612	290
	FSFS3	0.556885	0.579152	1.276174	0.110669	299
	MCFS	0.584998	0.669780	0.834086	0.111120	472
RHFCM	FSFS1	0.302229	0.962127	0.226512	0.212346	673
	FSFS2	0.439571	0.800167	0.373607	0.247220	788
	FSFS3	0.502227	0.571327	1.391428	0.121940	528
	MCFS	0.448492	0.959339	0.445315	0.297760	671
RH-IT2FCM	FSFS1	0.300205	0.780526	0.370741	0.194743	2104
	FSFS2	0.326650	0.861338	0.968081	0.198814	2177
	FSFS3	0.404447	0.670118	0.915298	0.082220	1987
	MCFS	0.523792	0.542756	1.492225	0.173367	2675
EM-SFSCL		0.604564	0.405678	1.213464	0.092346	4364
IT2FRH-SFSCL		**0.834504**	**0.172006**	**4.983455**	**0.043191**	2877

fuzzy approach helps to manage the uncertainty present in the fuzzy membership function.

The best performance of the proposed IT2FRH-SFSCL over other algorithms, in terms of different cluster validity indices, is achieved because (i) the simultaneous nature of feature selection and clustering helps to find out the best features necessary to group the objects in a data set while actually performing the grouping, (ii) the max-relevance max-significance [316] based feature selection maximizes both the relevance of the feature, as well as the significance of it with respect to the already-selected features, (iii) the rough hypercuboid approach, used in the IT2FRH-SFSCL, computes the lower approximation and boundary region implicitly for the clusters without any threshold value; (iv) the concept of crisp lower approximation and interval type-2 fuzzy boundary of the IT2FRH-SFSCL method deals with uncertainty, incompleteness, and vagueness in cluster definition; and (v) the IT2 fuzzy approach helps to manage the uncertainty present in the fuzzy membership function.

Table 7.5: Performance of Different Clustering Algorithms on GSE71107 Data Set

Different Algorithms		Silhouette Index	DB Index	Dunn Index	Xie-Beni Index	Time (msec)
HCM		0.377091	0.762593	1.584495	0.145465	651
FCM		0.343229	1.247167	0.590560	0.376418	2454
RFCM		0.359525	0.836476	0.882797	0.287403	864
RHFCM		0.471422	0.419962	2.619147	0.077129	217
RH-IT2FCM		0.734994	0.168680	5.383981	0.009920	1245
HCM	FSFS1	0.364181	1.078206	0.507135	0.420378	712
	FSFS2	0.391592	1.089672	0.574406	0.396444	783
	FSFS3	0.390873	0.996564	0.490344	0.534402	688
	MCFS	0.387999	1.098260	0.816458	0.261628	873
FCM	FSFS1	0.301146	1.312403	0.389541	0.562841	3561
	FSFS2	0.377505	1.158005	0.537067	0.450097	2871
	FSFS3	0.398028	1.027969	0.519841	0.517091	2897
	MCFS	0.411320	0.819176	1.170371	0.193519	3583
RFCM	FSFS1	0.373983	1.253587	0.705725	0.312340	784
	FSFS2	0.429619	1.157623	0.940403	0.296812	792
	FSFS3	0.418979	0.969706	0.813878	0.360198	889
	MCFS	0.400806	0.973321	0.889439	0.289646	982
RHFCM	FSFS1	0.403187	0.769959	0.615578	0.469700	385
	FSFS2	0.398047	3.458018	0.224909	1.213706	355
	FSFS3	0.409357	1.360076	0.660740	0.451934	487
	MCFS	0.351208	1.529741	0.578565	0.505390	488
RH-IT2FCM	FSFS1	0.382865	0.840743	0.533512	0.522770	984
	FSFS2	0.413898	2.021520	0.415954	0.698352	1158
	FSFS3	0.402663	1.491217	0.604475	0.491789	1273
	MCFS	0.358351	1.508067	0.615497	0.453404	1437
EM-SFSCL		0.731231	0.321312	5.365235	0.008756	2876
IT2FRH-SFSCL		**0.856123**	**0.092082**	**5.39167**	**0.002808**	1377

7.3.4 Functional Consistency of Clustering Results

DIANA microT [335] is one of the well known miRNA target prediction algorithms, which can be used to predict miRNA target genes for the miRNA clusters produced by different clustering algorithms. The functional consistency of the genes targeted by miRNAs of a cluster can be evaluated using the biological annotations of those genes of different clusters in terms of the gene ontology (GO). Hence, the GO Term Finder [60] can be used to calculate the annotation ratios of each targeted gene cluster in the three GO ontologies. The GO term is searched, to which, most of the genes of a particular cluster annotate [468].

The annotation ratio, also termed as cluster frequency, of all gene clusters for a particular ontology is found out, which is defined as the number of genes in both the assigned GO term and the cluster, divided by the number of genes in that cluster. The annotation ratio increases when the majority of genes in the cluster are functionally more close to each other and miRNAs targeting these genes are involved in common cellular processes. On the other hand, when the cluster contains more noise or irrelevant genes and the miRNAs targeting these genes are just randomly clustered, the annotation ratio decreases. The sum

of all annotation ratios is then calculated, which is known as the final annotation ratio. Increment in final annotation ratio indicates that the genes are better clustered than the other, providing a more functionally consistent clustering result [516].

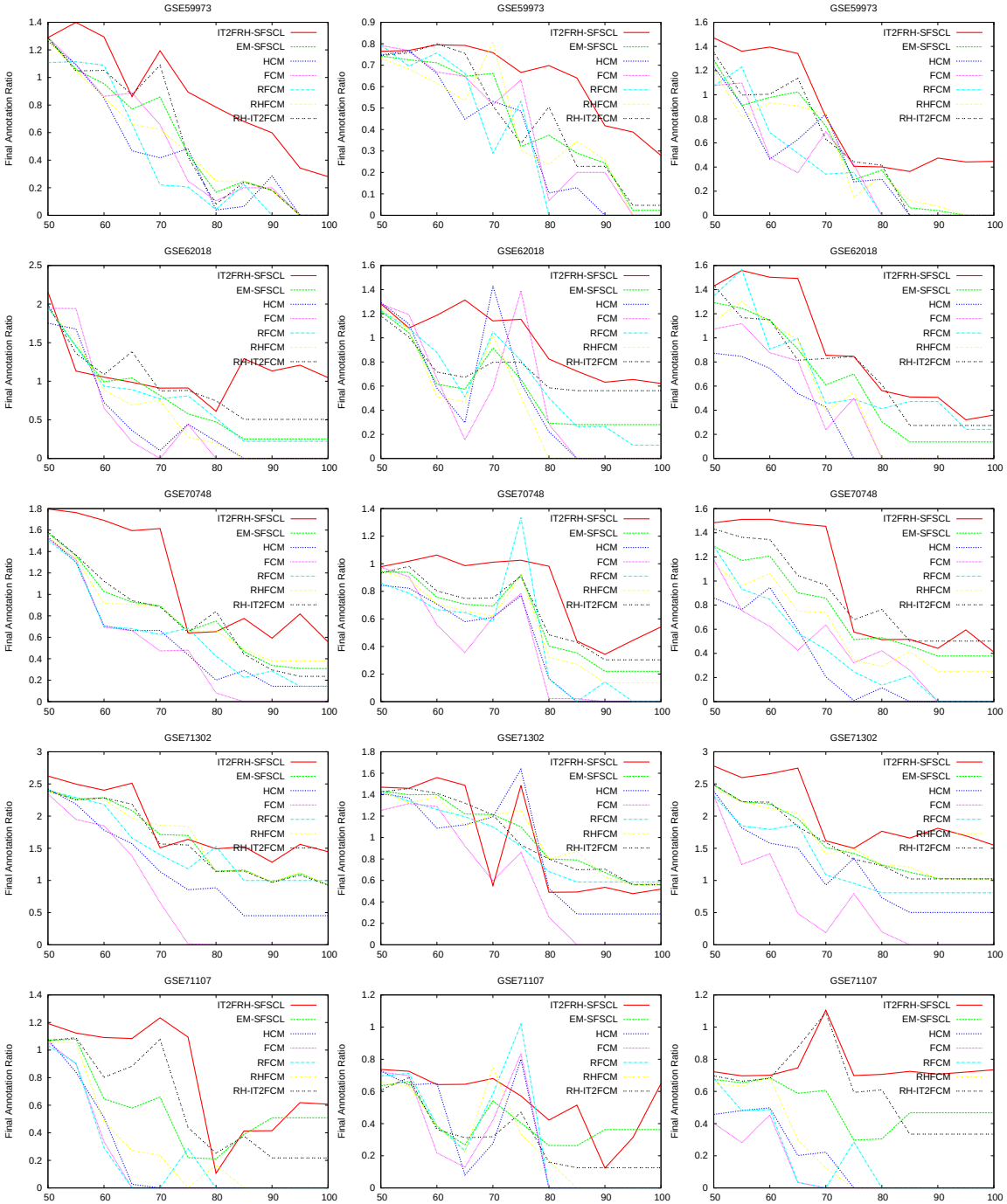
Further analysis were done on the genes which are targeted by minimum t percentage of miRNAs in each miRNA cluster, where t is varied from 50 to 100. Fig. 7.2 presents the comparative results of the HCM, FCM, RFCM, RHFCM, RH-IT2FCM, EM-SFSCL and the proposed algorithm with respect to the cluster frequency or final annotation ratio (FAR), for the Molecular Function (MF), Biological Process (BP), and Cellular Component (CC) ontologies on the five miRNA expression data sets mentioned in Section 7.3.1, where the horizontal axis represents t, the minimum percentage of miRNAs targeted in each miRNA cluster. All the results presented in the figure confirm that the proposed algorithm provides higher or comparable FARs in most of the cases than that produced by the other c-means algorithms. The experimental studies show that, for the MF, BP, and CC ontologies, the IT2FRH-SFSCL attains the highest FAR than that generated using HCM, FCM, and RFCM algorithms in 40, 50, and 52 cases, respectively, out of 55 comparisons each. The results also confirm that the proposed method provides higher or comparable final annotation ratios than that obtained using the RHFCM, RH-IT2FCM, and EM-SFSCL algorithms in most of the cases. In each of the 55 cases for the MF, BP, and CC ontologies, the proposed method provides a higher final annotation ratio in 44, 44, and 45 cases, respectively. Overall, the proposed method provides a higher final annotation ratio in 35, 42, and 44 cases for the MF, BP, and CC ontologies, respectively, out of 55 cases each.

Hence, the proposed IT2FRH-SFSCL algorithm can produce miRNA clusters which are functionally more compact than those obtained using other algorithms, while the other algorithms include irrelevant miRNAs, noise or outliers in the clusters.

7.3.5 Biologically Significant Gene Clusters

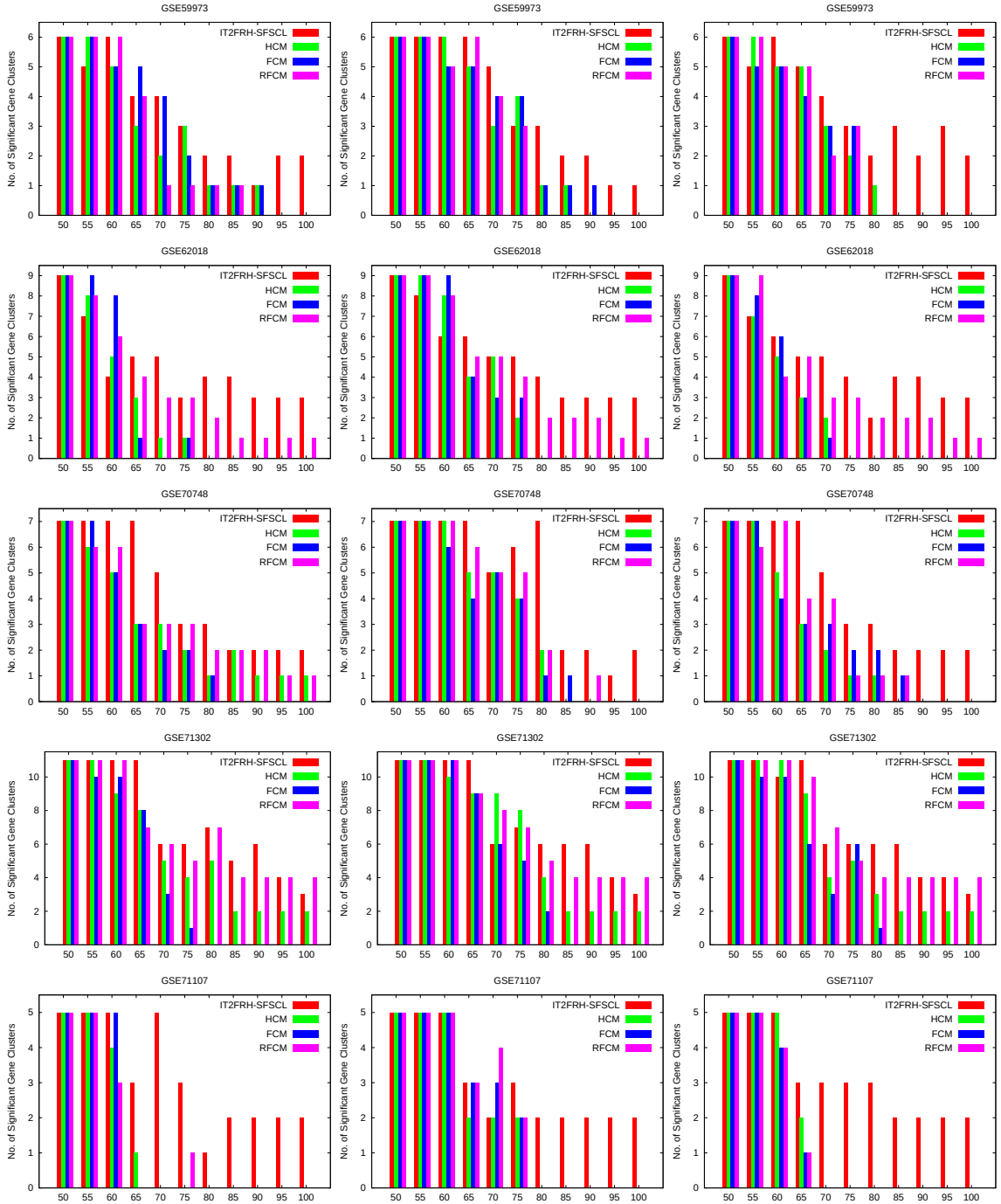
The genes that are targeted by at least 50% to 100% microRNAs are used to calculate the number of significant gene clusters. Figure 7.3 presents the results for the MF, BP, and CC ontologies for the data sets mentioned above. The GO Term Finder is used to determine the statistically significant gene clusters produced by different algorithms for all the GO terms from the MF, BP, and CC ontologies. If any cluster of genes generates a p-value smaller than 0.05, then that cluster is considered as a significant cluster. Figure 7.3 presents the comparative results of the HCM, FCM, and RFCM and the proposed algorithm for the MF, BP, and CC ontologies, respectively. From the results, it is seen that, out of 55 cases each, the proposed algorithm generates a higher or comparable number of significant gene clusters in 48, 50, and 50 cases, for the MF, BP, and CC ontologies, respectively.

Figure 7.4 shows the number of significant gene clusters generated by the RHFCM,



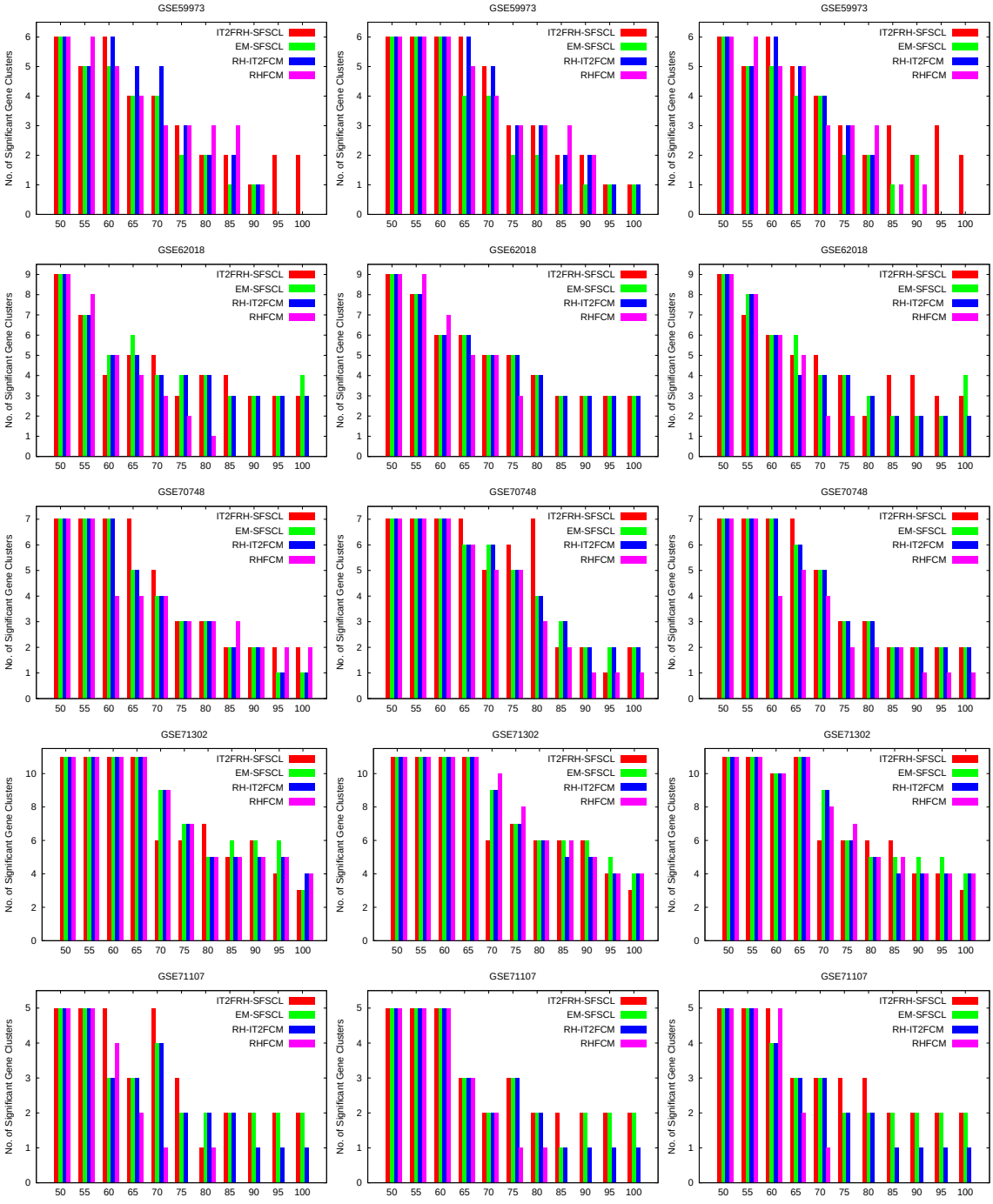
(a) Molecular Function (b) Biological Process (c) Cellular Component

Figure 7.2: Biological annotation ratios obtained using different algorithms with respect to percentage of genes targeted



(a) Molecular Function (b) Biological Process (c) Cellular Component

Figure 7.3: Biologically significant miRNA clusters obtained using HCM, FCM, RFCM, and the proposed algorithm



(a) Molecular Function (b) Biological Process (c) Cellular Component

Figure 7.4: Biologically significant miRNA clusters obtained using EM-SFSCL, RH-IT2FCM, RHFCM, and the proposed algorithm

RH-IT2FCM, EM-SFSCL, and proposed algorithm for the MF, BP, and CC ontologies for five data sets. All the results reported in the figure establish the fact that the proposed algorithm generates a higher or comparable number of significant gene clusters than that of the rough-fuzzy based c-means algorithms in most of the cases. For the three ontologies, the proposed method generates a higher or comparable number of significant gene clusters in 45, 38, and 44 cases, respectively. Overall, out of total 165 cases, it provides better results in 124 cases. So, from Figure 7.4, it can also be seen that the proposed clustering algorithm produces better results, irrespective of the ontologies and data sets used. Hence, it can be concluded that the proposed clustering technique generates highly compact and functionally enriched miRNA clusters.

7.3.6 Execution Time

Moreover, Tables 7.1 - 7.5 provide the execution time of different algorithms on five miRNA data sets. The results reported in Tables 7.1 - 7.5 establish the fact that the proposed algorithm provides good clustering results in lower or comparable time than most of the existing methods, irrespective of the data sets used. The lower execution time of the proposed algorithm is achieved due to its low computational complexity with the use of rough hypercuboid approach.

7.4 Conclusion

The contribution of the chapter is two-fold, namely,

1. development of a novel algorithm (IT2FRH-SFSCL), integrating the IT2FR-MRMS algorithm proposed in Chapter 4 and RH-IT2FCM algorithm proposed in Chapter 6, to simultaneously select the most important time points while grouping the miRNA data sets; and

2. demonstrating the effectiveness of the proposed algorithm, along with a comparison with existing clustering and feature selection, on several miRNA data sets using some well known cluster validity indices.

The proposed algorithm is formulated, combining the utility of both rough sets and IT2 fuzzy sets, for knowledge discovery tasks. The proposed algorithm provides the best performance in 82.45% cases for optimum parameter values. The proposed extension outperforms also other c-means algorithms, and feature selection-clustering combinations, irrespective of the miRNA data sets and quantitative indices used. The proposed algorithm performs better than any other c-means algorithms and their combinations in 97.69% cases, with respect to different cluster validity indices, and generates relevant clusters in lesser

or comparable time. It also shows better performance in 95.00% cases, when compared with expectation-maximization based simultaneous feature selection and clustering (EM-SFSCL) [271] algorithm. Moreover, the proposed algorithm performs significantly better than other methods, irrespective of the miRNA data sets and quantitative indices used, and provides biologically significant and relevant miRNA clusters. The proposed algorithm is also applied on some of the data sets used in Chapter 6, and the corresponding results are reported in Appendix B.

Chapter 8

Conclusion and Future Directions

This chapter concludes the thesis. A summary of the contributions presented in different chapters of this thesis is given, with a focus on the main contributions, namely, development of scalable dimensionality reduction and clustering algorithms, and their applications in various fields. It also provides future research directions, including possible extensions and applications of the proposed research work in feature selection and clustering of real life large data analysis.

8.1 Major Contributions

The thesis presents several methods for dimensionality reduction and clustering of large real life data sets. One of the major problems in real life data analysis is uncertainty. In this regard, the theories of rough sets and fuzzy sets (both type-1 and type-2) are judiciously integrated in the proposed research work to develop a strong mathematical framework for modeling and propagating the uncertainty inherent in real life noisy large data sets.

Chapter 3 presents a feature selection method, based on fuzzy-rough sets. It selects real valued noisy features by maximizing both relevance and significance of the selected features. Different feature evaluation criteria such as dependency, relevance, redundancy, and significance are presented in this chapter for attribute selection task using fuzzy-rough sets. The performance of different rough set models is compared with that of some existing feature evaluation indices based on the predictive accuracy of nearest neighbor rule, support vector machine, and decision tree. The effectiveness of the fuzzy-rough set based attribute selection method, along with a comparison with existing feature evaluation indices and different rough set models, has been demonstrated on a set of benchmark and microarray gene expression data. Although traditional type-1 fuzzy-rough sets, used in Chapter 3, is useful for feature selection in uncertain environment, it fails to provide good

performance when determination of exact membership function for any fuzzy set is difficult. In this regard, a new feature selection method, based on interval type-2 (IT2) fuzzy-rough sets, has been introduced in Chapter 4. Here, the features are also selected by maximizing both relevance and significance of the features. By introducing the concept of lower and upper fuzzy equivalence partition matrices, the lower and upper relevance and significance of the features have been defined for IT2 fuzzy approximation spaces. Different feature evaluation criteria have been presented for attribute selection task using IT2 fuzzy-rough sets. The performance of IT2 fuzzy-rough sets has been compared with that of some existing feature evaluation indices including classical rough sets, neighborhood rough sets, and type-1 fuzzy-rough sets. The effectiveness of the proposed IT2 fuzzy-rough set based attribute selection method, along with a comparison with existing feature selection and extraction methods, has been demonstrated on several real life data.

Feature extraction, another approach for reducing the dimension of a data set, generally provides a richer feature subset than that obtained using a feature selection algorithm, but with a higher cost. However, for a given data set, it is very difficult to decide whether to select a feature from original measurement space or extract a new feature by transforming the existing features. In this regard, a novel dimensionality reduction method, based on fuzzy-rough sets, has been introduced in Chapter 5, which simultaneously selects attributes and extracts features using the concept of feature significance. The feature set in each iteration is partitioned into three subsets, namely, insignificant, dispensable, and significant feature sets. The insignificant feature set is discarded from the current feature set, while the significant feature set is used to select or extract a feature in the next iteration. Depending on the quality of features present in the dispensable set of the current iteration, a new feature is extracted or an existing feature is selected from the dispensable set for a reduced feature set. In effect, the final reduced feature set may simultaneously contain some original features of the measurement space and extracted new features of the transformed space, which are both relevant and significant. The effectiveness of the proposed fuzzy-rough dimensionality reduction method, along with a comparison with other methods, has been demonstrated on a set of real life data using the predictive accuracy of different classifiers.

Chapter 6 presents a novel rough-fuzzy clustering algorithm. The proposed clustering algorithm integrates judiciously the theory of rough hypercuboid approach, interval type-2 fuzzy sets, and c-means algorithm, to approximately manage the uncertainty present in a data set. While the concept of lower approximation and boundary region of rough sets deals with uncertainty, incompleteness, and vagueness in cluster definition, the use of fuzzy membership of IT2 fuzzy sets in the boundary region enables efficient handling of overlapping partitions in uncertain environment. Using the concept of hypercuboid equivalence partition matrix of rough hypercuboid approach, the lower approximation and boundary region of each cluster are implicitly find out, without the need of any user

specified threshold. The interval valued fuzzifier has been used to deal with the uncertainty associated with the parameters of rough-fuzzy clustering algorithms, where determining the appropriate value of the fuzzifier is difficult. The problems of initial prototype selection and stuck in local minima of different c-means algorithms have been efficiently solved by developing a robust prototype selection method. The performance of the proposed method has been extensively compared with that of some existing clustering techniques, using several cluster validity indices on various real life data sets.

Finally, Chapter 7 presents an application of clustering, which is grouping functionally similar miRNAs from microarray data sets. In Chapter 7, the clustering algorithm proposed in Chapter 6 is judiciously combined with the feature selection algorithm proposed in Chapter 4 to develop an integrated method, which can select the optimum set of features while clustering the of miRNAs, keeping the complexity of the algorithm in mind. Comparisons are performed with other existing algorithms to prove the effectiveness of the proposed approach for grouping miRNA data sets.

In brief, the concept of simultaneous feature selection and feature extraction in dimensionality reduction is unique.

8.2 Future Directions

There are many important aspects of the research reported in this thesis that can be extended for the benefit of pattern recognition, data mining, fuzzy-rough sets, and big data analysis research community. Some of the probable future extensions are enlisted below.

1. The attribute selection methods, reported in Chapter 3 and Chapter 4, can be combined with fuzzy IF-THEN rules [523] and fuzzy decision tree [524] to deal with the data sets containing fuzzy linguistic variables directly without any discretization.

2. A feature selection method can be developed using rough-neural or neuro-fuzzy approach which can successfully be used for online [193, 518, 533] and time critical real life applications [563] and for mining big data [553].

3. The simultaneous feature selection and extraction method, proposed in Chapter 5, can be extended using interval type-2 fuzzy-rough sets and its effectiveness can be analysed in comparison with the complexity of the algorithm.

4. The rough-fuzzy clustering algorithm, proposed in Chapter 6, can be integrated with different types of distance measures and its effectiveness can be studied for real life large data sets.

5. Similar to the prototype initialization method, introduced in Chapter 6, some more robust methods can be developed that can lead to a quick convergence for the c-means clustering methods.

6. One of the important problems in clustering is the determination of the number of clusters, which clearly impacts and is influenced by the feature selection task. A technique may be developed which can simultaneously find out the optimum number of clusters while clustering the data set.

Appendix A

Bootstrap Smoothing and Different Quantitative Indices

Bootstrap Smoothing

In most of the cases, the cross validation error of any prediction rule $\mathcal{R}$ obtained during the feature selection process provides an under estimate (selection bias) of the prediction error rate of $\mathcal{R}$. To correct this selection bias, the cross-validation or bootstrap needs to be used external to the feature selection process. The variability of the leave-one-out error can be reduced by constructing an appropriate bootstrap procedure [115,116]. It provides a direct assessment of variability for estimated parameters in the prediction rule. Also, when the number of bootstrap K is set to be less than the number of objects n, the computation takes lesser time than leave-one-out cross-validation. The bootstrap smoothing [116] of leave-one-out cross-validation is a "smoothed" version of leave-one-out cross-validation error and is given by the leave-one-out bootstrap error $\mathcal{E}_{\text{BS}}$, which predicts the error at a point x_j only from bootstrap samples that do not contain the point x_j. The procedure to calculate $\mathcal{E}_{\text{BS}}$ is as follows:

Let K bootstrap samples of size n are obtained by resampling with replacement from the original set of n samples. Suppose the bootstrap version of the rule $\mathcal{R}$, $\mathcal{R}_k^\star$ is generated from the kth bootstrap sample, the procedure for generation is exactly the same as $\mathcal{R}$ was generated from the original training set using the same feature selection process. Monte Carlo algorithm gives an estimate of $\mathcal{E}_{\text{BS}}$ for the K bootstrap samples as

$$\mathcal{E}_{\text{BS}} = \frac{1}{n}\sum_{j=1}^{n}\mathcal{E}_j, \quad \text{where} \quad \mathcal{E}_j = \frac{\sum_{k=1}^{K} I_{jk}Q_{jk}}{\sum_{k=1}^{K} I_{jk}}; \tag{A.1}$$

and I_{jk} is 1 if x_j is not contained in the kth bootstrap sample and 0 otherwise, and Q_{jk} is 1 if $\mathcal{R}_k^\star$ misallocates x_j and 0 otherwise. Usually, $\mathcal{E}_{\text{BS}}$ gives the value based on approximately .632 of the number of original data points [115], and it was proved in [115] that it closely agrees with half-sample cross-validation while considering all the features. So, $\mathcal{E}_{\text{BS}}$ is upwardly biased. To correct the upward bias in $\mathcal{E}_{\text{BS}}$ with the downwardly biased apparent error (AE), which is denoted by the proportion of the original training samples misclassified, Efron [115] proposed the .632 estimator as follows:

$$\mathcal{E}_{\text{BS}}^{.632} = .368 \times AE + .632 \times \mathcal{E}_{\text{BS}}. \tag{A.2}$$

For the problems where there are large number of features compared to the number of samples, the constructed prediction rule can be over fitted. For highly overfitted rules like nearest neighbors, the AE is zero. In this scenario, the .632+ estimator $\mathcal{E}_{\text{BS}}^{.632}$ can be used [116], which puts relatively more weight on the leave-one-out bootstrap error $\mathcal{E}_{\text{BS}}$. The $\mathcal{E}_{\text{BS}}^{.632}$ estimator can be defined as

$$\mathcal{E}_{\text{BS}}^{.632} = (1 - w)AE + w\mathcal{E}_{\text{BS}}; \tag{A.3}$$

where the weight w is given by

$$w = \frac{.632}{1 - .368r}, \quad \text{and} \quad r = \frac{\mathcal{E}_{\text{BS}} - AE}{\gamma - AE} \tag{A.4}$$

is the relative overfitting rate. The γ is the no-information error rate that would apply if the distribution of the class-membership label of the jth sample did not depend on the corresponding feature vector x_j. γ can be estimated as

$$\gamma = \sum_{i=1}^{c} p_i(1 - q_i); \tag{A.5}$$

where p_i is the proportion of the original samples coming from the ith class, q_i is the proportion of them assigned to the ith class by $\mathcal{R}$, and c is the number of classes. The weight w ranges from .632 when $r = 0$ to 1 when $r = 1$. It may happen that $\gamma \leq AE$ or $AE < \gamma \leq \mathcal{E}_{\text{BS}}$, in that case r can fall outside the range $[0, 1]$. To account for this, the value of r must be truncated so that it falls into interval $[0, 1]$ [116].

Hence, in case when the amount of overfitting, measured by $(\mathcal{E}_{\text{BS}} - AE)$, is relatively large, $\mathcal{E}_{\text{BS}}^{.632}$ estimate puts more weight on the bootstrap leave-one-out error $\mathcal{E}_{\text{BS}}$ providing better estimate of the classification error.

Quantitative Indices

This section presents a brief description of different quantitative indices, which are used in several chapters of this thesis.

Class Separability Index

The class separability index $\mathcal{S}$ [101] of a data set is defined as

$$\mathcal{S} = \text{trace}(V_B^{-1} V_W); \tag{A.6}$$

where V_W is the within class scatter matrix and V_B is the between class scatter matrix, defined as follows:

$$V_W = \sum_{j=1}^{c} \pi_j E\{(X - v_j)(X - v_j)^T | \beta_j\} = \sum_{j=1}^{c} \pi_j \Sigma_j; \tag{A.7}$$

$$V_B = \sum_{j=1}^{c} \pi_j (v_j - \bar{v})(v_j - \bar{v})^T; \tag{A.8}$$

$$\text{and } \bar{v} = E\{X\} = \sum_{j=1}^{c} \pi_j v_j; \tag{A.9}$$

where c is the number of classes, π_j is a priori probability that a pattern belongs to class β_j, X is a feature vector, $\bar{v}$ is the sample mean vector for the entire data points, v_j and Σ_j represent the sample mean and covariance matrix of class β_j, respectively, and $E\{\cdot\}$ is the expectation operator. A lower value of class separability index $\mathcal{S}$ ensures that classes are well separated by their scatter means. Hence, a good feature subset should have the value of $\mathcal{S}$ index as low as possible.

Davies-Bouldin Index

Let $S(v_k)$ be the mean distance of the points belonging to cluster β_k to their center v_k:

$$S(v_k) = \frac{1}{n} \sum_{i \in I_k} ||x_{ik} - v_k||. \tag{A.10}$$

Also

$$d(v_k, v_{k'}) = ||v_k - v_{k'}|| \tag{A.11}$$

be the distance between the centers v_k and $v_{k'}$ of clusters β_k and $\beta_{k'}$.

The Davies-Bouldin (DB) index [94] is a function of the ratio of sum of within class distance to between class separation and is given by

$$\mathrm{DB} = \frac{1}{c}\sum_{k=1}^{c}\max_{k\neq k'}\left\{\frac{S(v_k)+S(v_{k'})}{d(v_k,v_{k'})}\right\} \quad \text{for } 1\leq k,k'\leq c. \tag{A.12}$$

The DB index minimizes the within class distance $S(v_k)$ and maximizes the between class separation $d(v_k, v_{k'})$, where v_k is the sample mean of class β_k. Therefore, for a given data set and c value, the higher the similarity values within the class and the between class separation, the lower would be the DB index value. A good feature set should have the value of DB index as low as possible.

Dunn Index

Dunn's index [50] is also designed to identify sets of clusters that are compact and well separated. Dunn's (D) index maximizes

$$\mathrm{D} = \min_{k}\left\{\min_{k\neq k'}\left\{\frac{d(v_k,v_{k'})}{\max_l S(v_l)}\right\}\right\} \quad \text{for } 1\leq k,k',l\leq c. \tag{A.13}$$

A good feature subset should have the value of Dunn index as high as possible.

Fuzzy Feature Evaluation Index

The fuzzy feature evaluation index is defined as [385]

$$\mathrm{FFEI} = \frac{2}{n(n-1)}\sum_{i}^{n}\sum_{j\neq i}^{n}\frac{1}{2}\Big[\mu_{ij}^{R}(1-\mu_{ij}^{O})+\mu_{ij}^{O}(1-\mu_{ij}^{R})\Big]; \tag{A.14}$$

where μ_{ij}^{O} and μ_{ij}^{R} are the degrees that both patterns x_i and x_j belong to the same cluster in the original feature space Ω_O and reduced feature space Ω_R, respectively, and n is the total number of samples. The membership function μ_{ij} can be defined as

$$\mu_{ij} = \begin{cases} 1-\frac{d_{ij}}{D} & \text{if } d_{ij}\leq D \\ 0 & \text{otherwise} \end{cases} \tag{A.15}$$

where d_{ij} is the distance between patterns x_i and x_j, and D may be expressed as

$$D = \beta d_{max}; \tag{A.16}$$

where d_{max} is the maximum separation between patterns in the respective feature spaces and β is an user defined constant ranging 0 to 1. In the present work, the value of β is

set to 0.2. d_{ij} can be defined in many ways, like Euclidean distance. The value of FFEI decreases as the intercluster (respectively, intracluster) distances increase (respectively, decrease). Hence, the lower the value of FFEI, the more crisp is the cluster structure.

Silhouette Index

To assess the quality of clusters, the Silhouette measure proposed by Rousseeuw [446] can be used. For computing the Silhouette value of an object x_i, first two scalars $a(x_i)$ and $b(x_i)$ are estimated. Let us note β_r the cluster to which object x_i belongs. The scalar $a(x_i)$ is the average distance between object x_i and all other objects of β_r. For any other cluster $\beta_s \neq \beta_r$, let $d(x_i, \beta_s)$ denote the average distance of object x_i to all objects of β_s. The scalar $b(x_i)$ is the smallest of these $d(x_i, \beta_s), r \neq s = 1, \cdots, c$. The Silhouette $s(x_i)$ of object x_i is then defined as

$$s(x_i) = \frac{b(x_i) - a(x_i)}{\max\{a(x_i), b(x_i)\}}. \tag{A.17}$$

The Silhouette value lies between -1 and 1. When its value is less than zero, the corresponding object is poorly classified.

Xie-Beni Index

The Xie-Beni index is an index of fuzzy clustering, but it is also applicable to crisp clustering. It is defined as the quotient between the mean quadratic error and the minimum of the minimal squared distances between the points in the clusters. The mean quadratic error, in the case of a crisp clustering, is expressed as

$$\frac{1}{N} \times WGSS; \tag{A.18}$$

where $WGSS$ denotes the within-cluster sum of squares. Xie-Beni index can be written as

$$XB = \frac{1}{N} \times \frac{WGSS}{\min_{i<j} \delta(C_i, C_j)^2}; \tag{A.19}$$

where,

$$\delta(C_i, C_j) = \min_{i \in k, j \in I_j} d(M_i, M_j). \tag{A.20}$$

Precision

Precision or Confidence (as it is called in data mining) denotes the proportion of predicted positive cases that are correctly. It can however analogously be called True Positive Accuracy (tpa), being a measure of accuracy of predicted positives in contrast with the rate of discovery of Real Positives (tpr). Precision is defined as:

$$Precision = Confidence = tpa = \frac{tp}{tp + fp}. \tag{A.21}$$

Sensitivity

Recall or Sensitivity is the proportion of real positive cases that are correctly predicted positive. In this context it is referred to as True Positive Rate (tpr). Recall is defined, with its various common appellations, by equation:

$$Sensitivity = Recall = tpr = \frac{tp}{tp + fn}. \tag{A.22}$$

Specificity

Inverse Recall or Specificity is thus the proportion of real negative cases that are correctly predicted negative, and is also known as the True Negative Rate (tnr).

$$Specificity = InverseRecall = tnr = \frac{tn}{tn + fp}. \tag{A.23}$$

Accuracy

Accuracy is effectively a prevalence-weighted average of Recall and Inverse Recall, as well as a bias-weighted average of Precision and Inverse Precision (also called True Negative Accuracy or tna).

$$Accuracy = tca = tcr = rp \times tpr + rn \times tnr = pp \times tpa + pn \times tna = \frac{tp + tn}{tp + fp + fn + tn}. \tag{A.24}$$

where, $rp = tp + fn$, $rn = fp + tn$, $pp = tp + fp$, and $pn = tn + fn$.

Jaccard Coefficient

The Jaccard index, also known as the Jaccard similarity coefficient [214], is a statistic used for comparing the similarity and diversity of sample sets. It measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the

union of the sample sets.

$$Jaccard = \frac{tp}{(tp + fn + fp)} = \frac{1}{(1 + 2 \times mean(fn, fp)/tp)} = \frac{F1}{(2 - F1)}. \tag{A.25}$$

where $F1$ represents the Dice Coefficient.

Dice Coefficient

The SørensenDice index, is a statistic used for comparing the similarity of two samples, which was independently developed by the botanists Thorvald Sørensen [477] and Lee Raymond Dice [104], published in 1948 and 1945 respectively. The SørensenDice is also known as F1 score or Dice similarity coefficient.

$$Dice = F1 = \frac{tp}{(tp + (fn + fp)/2)} = \frac{1}{(1 + mean(fn, fp)/tp)} = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \tag{A.26}$$

G Measure

While the F Measure (F1) is the harmonic mean of Recall and Precision, the G Measure is the geometric mean.

$$GMeasure = \sqrt{Precision \times Recall}. \tag{A.27}$$

This is also known as the FowlkesMallows index.

Matthews Correlation Coefficient

The Matthews Correlation Coefficient (MCC) can be calculated directly from the confusion matrix using the formula:

$$MCC = \frac{(tp \times tn) - (fp \times fn)}{\sqrt{(tp + fp)(tp + fn)(tn + fp)(tn + fn)}}. \tag{A.28}$$

Rand Index

Given a set of n elements $S = \{o_1, \cdots, o_n\}$ and two partitions of S to compare, $X = \{X_1, \cdots, X_r\}$, a partition of S into r subsets, and $Y = \{Y_1, \cdots, Y_s\}$, a partition of S into s subsets, define the following:

1. a, the number of pairs of elements in S that are in the same set in X and in the same set in Y

2. b, the number of pairs of elements in S that are in different sets in X and in different sets in Y

3. c, the number of pairs of elements in S that are in the same set in X and in different sets in Y

4. d, the number of pairs of elements in S that are in different sets in X and in the same set in Y

The Rand index, R, is:

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{\binom{n}{2}}. \tag{A.29}$$

Intuitively, $a+b$ can be considered as the number of agreements between X and Y and $c+d$ as the number of disagreements between X and Y.

Adjusted Rand Index

Given a set S of n elements, and two groupings (e.g. clusterings) of these points, namely $X = \{X_1, X_2, \ldots, X_r\}$ and $Y = \{Y_1, Y_2, \ldots, Y_s\}$, the overlap between X and Y can be summarized in a contingency table $[n_{ij}]$ where each entry n_{ij} denotes the number of objects in common between X_i and $Y_j : n_{ij} = |X_i \cap Y_j|$. Suppose, $a_i = \sum_{j=1}^{s} n_{ij}$ and $b_j = \sum_{i=1}^{r} n_{ij}$.

The Adjusted Rand Index is the corrected-for-chance version of the Rand Index. Though the Rand Index may only yield a value between 0 and +1, the Adjusted Rand Index can yield negative values if the index is less than the expected index. It is defined as:

$$Adjusted\ Index = \frac{Index - Expected\ Index}{Max\ Index - Expected\ Index}; \tag{A.30}$$

more specifically,

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}. \tag{A.31}$$

Appendix B

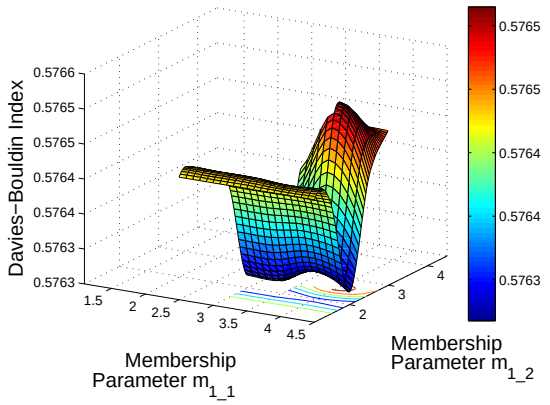
Supplementary Results on Different Data Sets

The performance of the IT2FRH-SFSCL algorithm proposed in Section 7.2 of Chapter 7 is also compared extensively with that of different existing clustering algorithms and feature selection-clustering combinations. In the later version, feature selection techniques are applied first on the data sets to select the most important features, and then clustering techniques are used to partition the data sets. As no class information is present before clustering, only unsupervised feature selection algorithms can be used for this purpose. To compare the clustering performance, the proposed technique is compared with the algorithm mentioned in the Section 7.3 of Chapter 7.

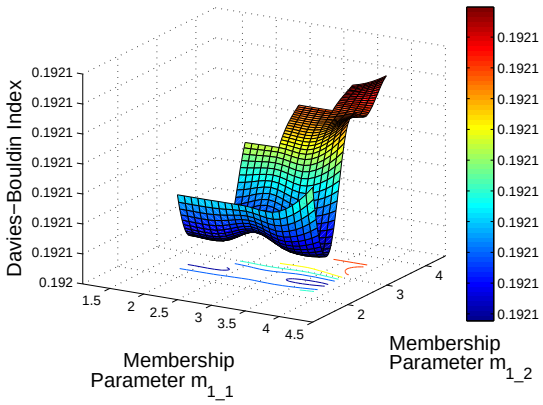
The experimental results on five data sets are presented in this section, namely, Glass, Hepatitis, Wine, Wisconsin Breast Cancer (WBC), and Echocardiogram, which are downloaded from *UCI Machine Learning Repository* (https://archive.ics.uci.edu/ml/datasets.html). The description of the data sets are provided in Section 6.4.2 of Chapter 6. Subsequent discussions analyze the results with respect to ten different cluster validity indices, namely, precision [424], sensitivity [424], specificity [424], accuracy [424], Jaccard Coefficient [424], Dice Coefficient [104], G Measure [275], Matthews Correlation Coefficient [339], Rand index [433], Adjusted Rand index [205] and execution time.

Optimum Values of Different Parameters

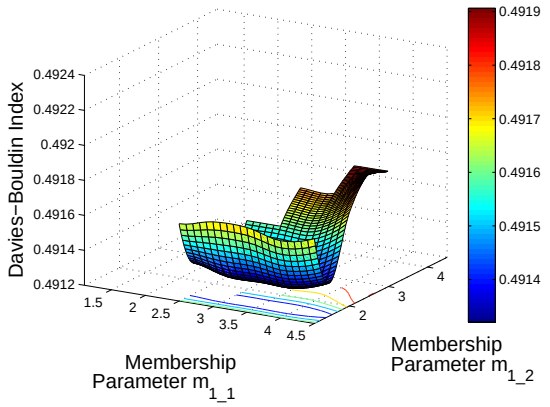
The parameters $\acute{m}_1$, $\acute{m}_2$, λ_1, and λ_2 are optimized using the method discussed in Section 6.5.1 of Chapter 6. The optimum values of $\acute{m}_1^\star$, $\acute{m}_2^\star$, $\lambda_1^\star$, and $\lambda_2^\star$, obtained using (6.42) of Chapter 6, are {1.5, 2.0, 0.94, 0.92}, {1.5, 2.0, 0.96, 0.86}, {1.5, 2.0, 0.96, 0.88}, {1.5, 2.0, 0.92, 0.94}, and {1.1, 2.0, 0.96, 0.86}, respectively, for Glass, Hepatitis, Wine, WBC, and Echocardiogram data sets.



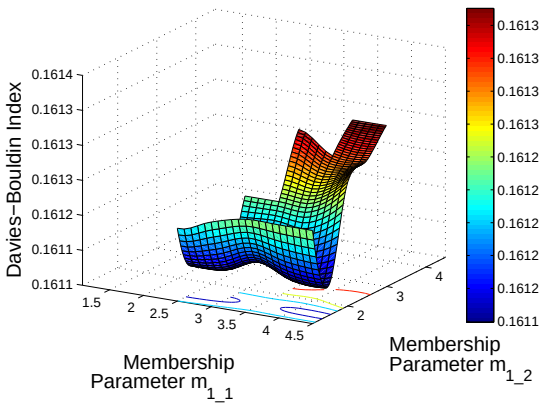
(a) Glass ($\lambda_1^\star = 0.94, \lambda_2^\star = 0.92$)



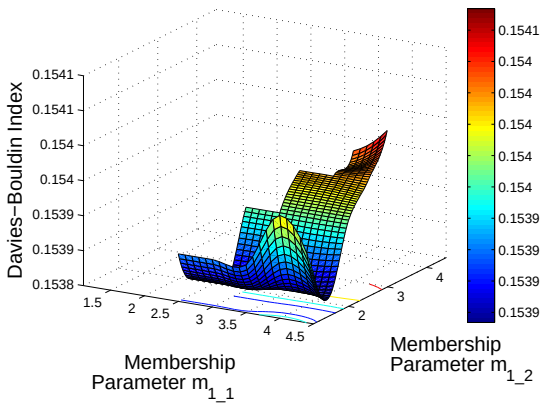
(b) Hepatitis ($\lambda_1^\star = 0.96, \lambda_2^\star = 0.86$)



(c) Wine ($\lambda_1^\star = 0.96, \lambda_2^\star = 0.88$)

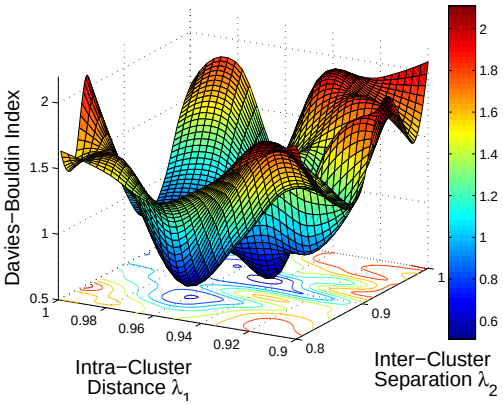


(d) WBC ($\lambda_1^\star = 0.92, \lambda_2^\star = 0.94$)

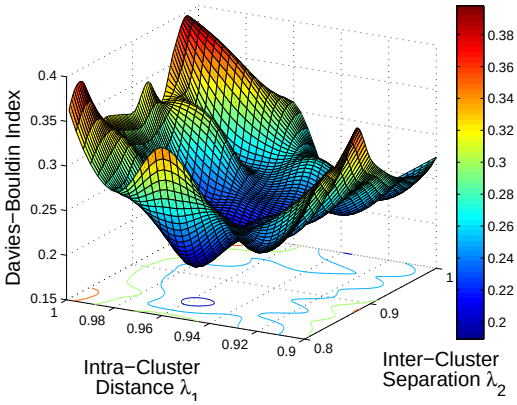


(e) Echocardiogram ($\lambda_1^\star = 0.96, \lambda_2^\star = 0.86$)

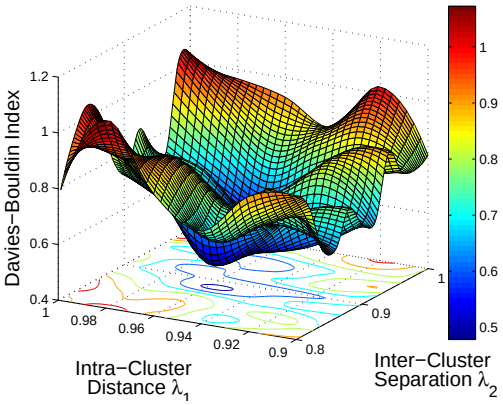
Figure B.1: Variation of DB index for different values of $\acute{m}_1$ and $\acute{m}_2$



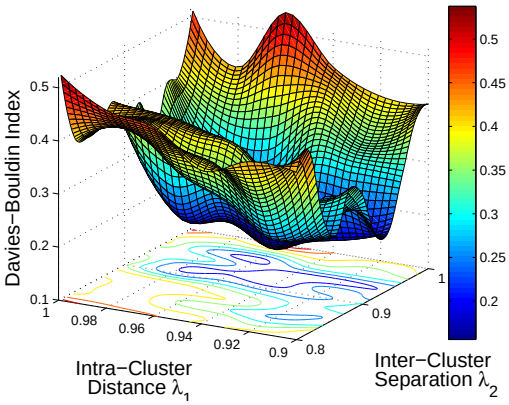
(a) Glass ($\acute{m}_1^\star = 2.5, \acute{m}_2^\star = 4.0$)



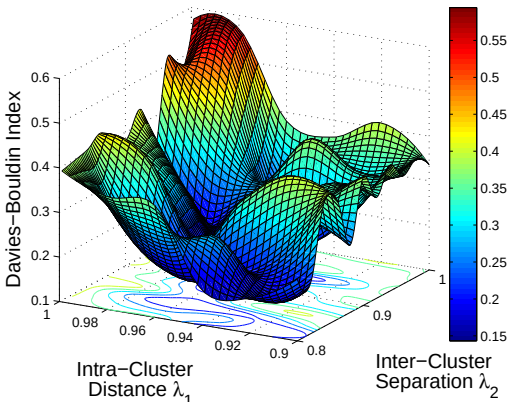
(b) Hepatitis ($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)



(c) Wine ($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)



(d) WBC ($\acute{m}_1^\star = 2.0, \acute{m}_2^\star = 2.5$)



(e) Echocardiogram($\acute{m}_1^\star = 1.5, \acute{m}_2^\star = 2.0$)

Figure B.2: Variation of DB index for different values of λ_1 and λ_2

In this regard, it should be noticed that the IT2FRH-SFSCL algorithm proposed in the Section 7.2 of Chapter 7 delivers best performance at $\acute{m}_1 = \acute{m}_1^\star$, $\acute{m}_2 = \acute{m}_2^\star$, $\lambda_1 = \lambda_1^\star$, $\lambda_2 = \lambda_2^\star$ in 45 cases, out of 50, with respect to different cluster validity indices. However, the precision provides the maximum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.92$, $\lambda_2 = 0.86$ for Hepatitis data, accuracy provides the maximum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.92$, $\lambda_2 = 0.86$ for Wine data, and Jaccard Coefficient attains its minimum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.0$, $\lambda_1 = 0.94$, $\lambda_2 = 0.86$, Dice Coefficient attains its minimum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.5$, $\lambda_1 = 0.92$, $\lambda_2 = 0.82$, G Measure attains its minimum value at $\acute{m}_1 = 1.5$, $\acute{m}_2 = 2.5$, $\lambda_1 = 0.92$, $\lambda_2 = 0.86$ for Echocardiogram data. Also, all other clustering algorithms achieve their best performance at $\acute{m}_1 = \acute{m}_1^\star$, $\acute{m}_2 = \acute{m}_2^\star$, $\lambda_1 = \lambda_1^\star$ and $\lambda_2 = \lambda_2^\star$ in 1154 cases, out of 1350 cases, irrespective of the cluster validity indices used. Hence, all the results reported above establish the importance of proposed parameter optimization technique, irrespective of the clustering algorithms compared.

Performance Analysis

The performance of the proposed IT2FRH-SFSCL algorithm is compared with that of some existing c-means algorithms on five data sets with respect to several cluster validity indices. Tables B.1 - B.5 present the comparative assessment of these five clustering algorithms and four unsupervised feature selection algorithms, in terms of precision, sensitivity, specificity, accuracy, Jaccard Coefficient, Dice Coefficient, G Measure, Matthews Correlation Coefficient, Rand index, and Adjusted Rand index index. All the results correspond to optimum values of λ_1 and λ_2. The bold values in these tables signify the best values. Out of 1300 cases, in 1291 cases, the IT2FRH-SFSCL algorithm with proposed centroid initialization method discussed in Section 6.3.6 of Chapter 6 provides better results than that of other clustering algorithms, in terms of different cluster validity indices.

Tables B.1 - B.5 provide the comparative results of different feature selection and clustering algorithms with the proposed centroid initialization method for five data sets. The results are analyzed to prove the performance increment due to simultaneous feature selection and clustering over the traditional approach where those were done serially. Out of 200 cases each for Glass, Hepatitis, Wine, Wisconsin Breast Cancer, and Echocardiogram data sets, the IT2FR-SFSCL algorithm proposed in the Section 7.2 of Chapter 7 performs better in lesser or comparable time in each and every cases, except the Echocardiogram (where it is 192), when compared with the other feature selection-clustering combinations, where they are performed in serial fashion. The best performance of the IT2FRH-SFSCL over the other serial algorithm combinations, in terms of the various cluster validity indices, is achieved because the simultaneous nature of feature selection and clustering helps to find out the best features necessary to group the objects in a data set while actually

Table B.1: Performance of Different Clustering Algorithms on Glass Data Sets using Different Cluster Validity Indices

Different Algorithms		Precision	Sensitivity	Specificity	Accuracy	Matthews Corr Coeff	Jaccard Coefficient	Dice Coefficient	G Measure	Rand Index	Adjusted Index	Time (msec)
HCM		0.370074	0.399442	0.876769	0.816199	0.260748	0.237775	0.384198	0.384478	0.698214	0.191585	56
FCM		0.398862	0.434217	0.877765	0.800623	0.287251	0.262458	0.415789	0.416164	0.711816	0.158039	178
RFCM		0.370157	0.407499	0.868287	0.799065	0.249593	0.240642	0.387931	0.388379	0.689790	0.150110	76
RHFCM		0.413290	0.437394	0.887096	0.841121	0.314407	0.269842	0.425001	0.425171	0.676363	0.265376	47
RH-IT2FCM		0.426980	0.451680	0.894342	0.850467	0.335526	0.281216	0.438983	0.439156	0.686367	0.270062	117
HCM	FSFS1	0.300039	0.208127	0.842269	0.767913	0.085378	0.140102	0.245771	0.249892	0.596771	0.029249	72
	FSFS2	0.357224	0.313386	0.863735	0.786604	0.185242	0.200388	0.333872	0.334588	0.658769	0.080154	89
	FSFS3	0.320721	0.322129	0.866996	0.781931	0.185814	0.191485	0.321423	0.321424	0.687420	0.106142	44
	MCFS	0.344228	0.366977	0.873934	0.808411	0.227723	0.215982	0.355239	0.355421	0.684876	0.168900	45
FCM	FSFS1	0.409204	0.463126	0.875091	0.800623	0.295875	0.277545	0.434498	0.435331	0.690580	0.115282	406
	FSFS2	0.335334	0.335418	0.861854	0.778816	0.193955	0.201473	0.335376	0.335376	0.686367	0.097254	281
	FSFS3	0.324392	0.330069	0.868410	0.785047	0.193140	0.195604	0.327206	0.327218	0.691984	0.117062	180
	MCFS	0.365265	0.366023	0.885952	0.809969	0.246097	0.223724	0.365644	0.365644	0.715589	0.181564	343
RFCM	FSFS1	0.201195	0.197164	0.842687	0.771028	0.044058	0.110592	0.199159	0.199169	0.601817	0.035376	68
	FSFS2	0.353287	0.319389	0.865009	0.799065	0.200436	0.201550	0.335484	0.335911	0.660348	0.108425	60
	FSFS3	0.280349	0.314958	0.869524	0.800623	0.165947	0.174155	0.296647	0.297150	0.676363	0.151732	68
	MCFS	0.310576	0.352516	0.867707	0.789720	0.194063	0.197763	0.330220	0.330882	0.689044	0.122056	98
RHFCM	FSFS1	0.313183	0.208691	0.841003	0.767913	0.088397	0.143168	0.250476	0.255653	0.585056	0.016468	62
	FSFS2	0.282233	0.231761	0.847325	0.792835	0.101007	0.145816	0.254519	0.255755	0.549033	0.060250	75
	FSFS3	0.317083	0.321137	0.870519	0.794393	0.183249	0.189837	0.319097	0.319104	0.682682	0.171223	52
	MCFS	0.344228	0.366977	0.873934	0.808411	0.227723	0.215982	0.355239	0.355421	0.684876	0.168900	84
RH-IT2FCM	FSFS1	0.316406	0.208691	0.840696	0.767913	0.089133	0.143838	0.251501	0.256965	0.583344	0.017273	227
	FSFS2	0.282233	0.231761	0.847325	0.792835	0.101007	0.145816	0.254519	0.255755	0.549033	0.060250	272
	FSFS3	0.317083	0.321137	0.870519	0.794393	0.183249	0.189837	0.319097	0.319104	0.682682	0.171223	162
	MCFS	0.344228	0.366977	0.873934	0.808411	0.227723	0.215982	0.355239	0.355421	0.684876	0.168900	775
EM-SFSCL		0.336301	0.361194	0.877414	0.814815	0.224932	0.210876	0.348303	0.348525	0.683692	0.178745	2423
IT2FRH-SFSCL		**0.644422**	**0.610298**	**0.903920**	**0.856874**	**0.519672**	**0.456554**	**0.626896**	**0.627128**	**0.798956**	**0.289075**	2111

Table B.2: Performance of Different Clustering Algorithms on Hepatitis Data Sets using Different Cluster Validity Indices

Different Algorithms		Precision	Sensitivity	Specificity	Accuracy	Matthews Corr Coeff	Jaccard Coefficient	Dice Coefficient	G Measure	Rand Index	Adjusted Index	Time (msec)
HCM		0.402685	0.491803	0.491803	0.794702	-0.056486	0.284361	0.442805	0.445019	0.688312	0.159294	11
FCM		0.522351	0.503430	0.503430	0.780645	0.017511	0.344733	0.512716	0.512803	0.655300	0.006668	151
RFCM		0.599714	0.579649	0.579649	0.754839	0.178237	0.417948	0.589511	0.589596	0.627482	0.111920	20
RHFCM		0.522351	0.503430	0.503430	0.780645	0.017511	0.344733	0.512716	0.512803	0.655300	0.006668	9
RH-IT2FCM		0.745448	0.767149	0.767149	0.832258	0.512138	0.607902	0.756143	0.756221	0.718978	0.378243	17
HCM	FSFS1	0.676456	0.763338	0.763338	0.716129	0.431127	0.559182	0.717276	0.718585	0.590783	0.177178	44
	FSFS2	0.443121	0.440549	0.440549	0.625806	-0.116302	0.283558	0.441831	0.441833	0.528613	-0.054245	33
	FSFS3	0.415435	0.421494	0.421494	0.632258	-0.162958	0.264577	0.418443	0.418454	0.531965	-0.078395	25
	MCFS	0.711038	0.792429	0.792429	0.780645	0.496844	0.599399	0.749530	0.750631	0.655300	0.289795	51
FCM	FSFS1	0.666417	0.753938	0.753938	0.664516	0.411143	0.547366	0.707481	0.708828	0.551236	0.104439	114
	FSFS2	0.459136	0.451474	0.451474	0.606452	-0.089062	0.294727	0.455273	0.455289	0.519564	-0.037423	54
	FSFS3	0.387592	0.360518	0.360518	0.535484	-0.250430	0.229683	0.373565	0.373810	0.499288	-0.067401	68
	MCFS	0.666417	0.753938	0.753938	0.664516	0.411143	0.547366	0.707481	0.708828	0.551236	0.104439	101
RFCM	FSFS1	0.631432	0.667429	0.667429	0.729032	0.296685	0.480310	0.648932	0.649181	0.602346	0.158943	31
	FSFS2	0.457058	0.487170	0.487170	0.754839	-0.046945	0.308587	0.471634	0.471874	0.627482	-0.025288	123
	FSFS3	0.385714	0.439024	0.439024	0.696774	-0.166957	0.258373	0.410646	0.411507	0.574696	-0.091402	26
	MCFS	0.783410	0.783410	0.783410	0.858065	0.566819	0.643939	0.783410	0.783410	0.754839	0.445405	87
RHFCM	FSFS1	0.655836	0.721164	0.721164	0.722581	0.371296	0.523171	0.686950	0.687725	0.596481	0.175007	25
	FSFS2	0.409392	0.392403	0.392403	0.567742	-0.197475	0.250561	0.400718	0.400807	0.505991	-0.066733	29
	FSFS3	0.417604	0.425559	0.425559	0.638710	-0.156635	0.267061	0.421544	0.421563	0.535484	-0.077224	27
	MCFS	0.677427	0.770198	0.770198	0.690323	0.437907	0.563526	0.720840	0.722325	0.569669	0.140637	45
RH-IT2FCM	FSFS1	0.655899	0.727836	0.724252	0.725584	0.374897	0.526714	0.689997	0.690932	0.596481	0.175007	37
	FSFS2	0.410988	0.395895	0.395890	0.569042	-0.175395	0.252583	0.403300	0.403371	0.505991	-0.066733	48
	FSFS3	0.418891	0.427504	0.427570	0.644556	-0.127154	0.268355	0.423154	0.423176	0.535484	-0.077224	31
	MCFS	0.720455	0.808054	0.808054	0.787097	0.521198	0.615175	0.761744	0.762998	0.662673	0.306860	51
EM-SFSCL		0.726054	0.745833	0.745833	0.815789	0.471472	0.582042	0.735811	0.735877	0.697456	0.578845	406
IT2FRH-SFSCL		**0.844945**	**0.888393**	**0.888393**	**0.907143**	**0.732049**	**0.763861**	**0.866124**	**0.866397**	**0.861667**	**0.722203**	276

Table B.3: Performance of Different Clustering Algorithms on Wine Data Sets using Different Cluster Validity Indices

Different Algorithms		Precision	Sensitivity	Specificity	Accuracy	Matthews Corr Coeff	Jaccard Coefficient	Dice Coefficient	G Measure	Rand Index	Adjusted Index	Time (msec)
HCM		0.398338	0.652778	0.805082	0.726592	0.396769	0.328694	0.494762	0.509928	0.868596	0.580800	19
FCM		0.500234	0.651163	0.865992	0.829574	0.483717	0.394511	0.565806	0.570731	0.942741	0.662162	49
RFCM		0.406005	0.581084	0.769586	0.689139	0.300088	0.314076	0.478018	0.485719	0.910874	0.538049	50
RHFCM		0.398338	0.652778	0.805082	0.726592	0.396769	0.328694	0.494762	0.509928	0.868596	0.580800	30
RH-IT2FCM		0.953396	0.957747	0.975027	0.966292	0.929910	0.914915	0.955567	0.955569	0.954294	0.845674	23
HCM	FSFS1	0.431856	0.518103	0.766573	0.711610	0.267309	0.308099	0.471064	0.473018	0.726465	0.396859	40
	FSFS2	0.540686	0.553216	0.785784	0.737828	0.349512	0.376348	0.546879	0.546915	0.714086	0.400724	39
	FSFS3	0.794643	0.713615	0.831028	0.771536	0.578971	0.602502	0.751952	0.753040	0.970545	0.822101	28
	MCFS	0.960902	0.958478	0.979847	0.973783	0.939821	0.922500	0.959688	0.959689	0.956199	0.877803	32
FCM	FSFS1	0.835640	0.689928	0.836493	0.794038	0.582521	0.607491	0.755825	0.759297	0.883768	0.579354	72
	FSFS2	0.831562	0.679870	0.823259	0.774194	0.558067	0.597577	0.748104	0.751900	0.872723	0.570834	68
	FSFS3	0.858170	0.707317	0.866200	0.835616	0.629268	0.633286	0.775475	0.779101	0.867756	0.563423	71
	MCFS	0.947887	0.949088	0.974718	0.966292	0.922919	0.902021	0.948487	0.948487	0.942614	0.839195	76
RFCM	FSFS1	0.847166	0.860421	0.925566	0.898876	0.778296	0.744808	0.853742	0.853768	0.894877	0.674183	31
	FSFS2	0.873070	0.881646	0.938861	0.917603	0.815237	0.781479	0.877337	0.877348	0.908462	0.724458	32
	FSFS3	0.454312	0.599667	0.776721	0.700375	0.340166	0.348587	0.516966	0.521954	0.928204	0.568231	37
	MCFS	0.956201	0.951533	0.976731	0.970037	0.931158	0.911792	0.953861	0.953864	0.949343	0.859028	33
RHFCM	FSFS1	0.431856	0.518103	0.766573	0.711610	0.267309	0.308099	0.471064	0.473018	0.726465	0.396859	27
	FSFS2	0.540686	0.553216	0.785784	0.737828	0.349512	0.376348	0.546879	0.546915	0.714086	0.400724	29
	FSFS3	0.794643	0.713615	0.831028	0.771536	0.578971	0.602502	0.751952	0.753040	0.970545	0.822101	37
	MCFS	0.946796	0.924095	0.963168	0.955056	0.900929	0.878478	0.935308	0.935377	0.924078	0.795055	31
RH-IT2FCM	FSFS1	0.813086	0.827753	0.908528	0.876405	0.728242	0.695424	0.820354	0.820387	0.866565	0.585571	36
	FSFS2	0.855817	0.860957	0.928450	0.902622	0.785026	0.751895	0.858379	0.858383	0.864026	0.615100	31
	FSFS3	0.914352	0.925758	0.958149	0.943820	0.877148	0.851886	0.920020	0.920037	0.943630	0.832168	41
	MCFS	0.946796	0.924095	0.963168	0.955056	0.900929	0.878478	0.935308	0.935377	0.924078	0.795055	37
EM-SFSCL		0.947887	0.949088	0.974718	0.966292	0.922919	0.902021	0.948487	0.948487	0.942614	0.839195	232
IT2FRH-SFSCL		**0.989418**	**0.981982**	**0.992424**	**0.991051**	**0.978927**	**0.971776**	**0.985686**	**0.985693**	**0.989675**	**0.940675**	117

Table B.4: Performance of Different Clustering Algorithms on WBC Data Sets using Different Cluster Validity Indices

Different Algorithms		Precision	Sensitivity	Specificity	Accuracy	Matthews Corr Coeff	Jaccard Coefficient	Dice Coefficient	G Measure	Rand Index	Adjusted Index	Time (msec)
HCM		0.916869	0.894529	0.894529	0.912127	0.811090	0.827420	0.905561	0.905630	0.839414	0.676505	108
FCM		0.919231	0.876989	0.876989	0.903339	0.795098	0.814245	0.897613	0.897862	0.842323	0.682865	181
RFCM		0.860606	0.674528	0.674528	0.757469	0.501741	0.608091	0.756289	0.761907	0.631934	0.237485	51
RHFCM		0.919231	0.876989	0.876989	0.903339	0.795098	0.814245	0.897613	0.897862	0.825058	0.646588	48
RH-IT2FCM		0.912882	0.901677	0.901677	0.913884	0.814482	0.830236	0.907245	0.907262	0.867456	0.688675	57
HCM	FSFS1	0.924507	0.868440	0.868440	0.899824	0.790962	0.810933	0.895597	0.896035	0.819401	0.634476	42
	FSFS2	0.806840	0.771220	0.771220	0.806678	0.576962	0.651020	0.788628	0.788829	0.687554	0.368229	56
	FSFS3	0.787553	0.767606	0.767606	0.796134	0.554801	0.635928	0.777452	0.777516	0.674819	0.344314	109
	MCFS	0.946680	0.913251	0.913251	0.933216	0.859281	0.868574	0.929665	0.929815	0.875133	0.748140	54
FCM	FSFS1	0.925917	0.876473	0.876473	0.905097	0.800865	0.819037	0.900517	0.900856	0.827904	0.652019	62
	FSFS2	0.786044	0.796271	0.796271	0.799649	0.582226	0.654429	0.791124	0.791141	0.679014	0.356815	56
	FSFS3	0.770369	0.781750	0.781750	0.783831	0.552002	0.634011	0.776018	0.776039	0.660524	0.320021	207
	MCFS	0.939637	0.924674	0.924674	0.936731	0.864181	0.872826	0.932095	0.932125	0.881259	0.761080	174
RFCM	FSFS1	0.915908	0.843897	0.843897	0.882250	0.756385	0.783213	0.878429	0.879166	0.791864	0.577701	58
	FSFS2	0.787966	0.531618	0.531618	0.650264	0.190840	0.465086	0.634892	0.647222	0.544358	0.033853	83
	FSFS3	0.737901	0.531176	0.531176	0.648506	0.172241	0.446865	0.617701	0.626063	0.543306	0.032914	74
	MCFS	0.943620	0.902859	0.902859	0.926186	0.845497	0.856648	0.922790	0.923015	0.863029	0.723443	77
RHFCM	FSFS1	0.908467	0.811321	0.811321	0.859402	0.713202	0.750011	0.857150	0.858521	0.757915	0.506901	78
	FSFS2	0.791420	0.736728	0.736728	0.783831	0.525308	0.616939	0.763095	0.763584	0.660524	0.310615	67
	FSFS3	0.764825	0.754558	0.754558	0.778559	0.519282	0.612457	0.759657	0.759674	0.654583	0.304783	75
	MCFS	0.944793	0.905218	0.905218	0.927944	0.849089	0.859742	0.924582	0.924794	0.866036	0.729563	68
RH-IT2FCM	FSFS1	0.908467	0.811321	0.811321	0.859402	0.713202	0.750011	0.857150	0.858521	0.757915	0.506901	107
	FSFS2	0.791420	0.736728	0.736728	0.783831	0.525308	0.616939	0.763095	0.763584	0.660524	0.310615	261
	FSFS3	0.764825	0.754558	0.754558	0.778559	0.519282	0.612457	0.759657	0.759674	0.654583	0.304783	286
	MCFS	0.944793	0.905218	0.905218	0.927944	0.849089	0.859742	0.924582	0.924794	0.866036	0.729563	3303
EM-SFSCL		0.938529	0.925632	0.925632	0.936731	0.864065	0.872722	0.932036	0.932058	0.881259	0.765385	2126
IT2FRH-SFSCL		**0.955060**	**0.942522**	**0.942522**	**0.952212**	**0.897494**	**0.902497**	**0.948750**	**0.948770**	**0.914045**	**0.825798**	667

Table B.5: Performance of Different Clustering Algorithms on Echocardiogram Data Sets using Different Cluster Validity Indices

Different Algorithms		Precision	Sensitivity	Specificity	Accuracy	Matthews Corr Coeff	Jaccard Coefficient	Dice Coefficient	G Measure	Rand Index	Adjusted Index	Time (msec)
HCM		0.445833	0.494767	0.494767	0.552846	-0.033674	0.306359	0.469027	0.469663	0.501695	0.021554	28
FCM		0.521654	0.502582	0.502582	0.541985	0.014955	0.344032	0.511940	0.512029	0.499706	-0.000579	51
RFCM		0.536667	0.536150	0.536150	0.541985	0.072815	0.366501	0.536408	0.536408	0.504639	0.009342	43
RHFCM		0.557215	0.550822	0.550822	0.564885	0.107847	0.383126	0.554000	0.554009	0.499706	-0.000070	5
RH-IT2FCM		0.566964	0.566667	0.566667	0.566667	0.133631	0.395493	0.566815	0.566815	0.512624	0.021346	16
HCM	FSFS1	0.582609	0.575822	0.575822	0.587786	0.158285	0.407654	0.579196	0.579206	0.511685	0.023417	53
	FSFS2	0.522201	0.522066	0.522066	0.526718	0.044267	0.353302	0.522133	0.522133	0.497592	-0.004812	54
	FSFS3	0.591834	0.581573	0.581573	0.595420	0.173103	0.415087	0.586659	0.586681	0.514504	0.029071	45
	MCFS	0.591214	0.582864	0.582864	0.595420	0.173878	0.415437	0.587009	0.587024	0.514504	0.029060	34
FCM	FSFS1	0.574215	0.568779	0.568779	0.580153	0.142891	0.400054	0.571484	0.571491	0.509102	0.018245	52
	FSFS2	0.534810	0.533568	0.533568	0.541985	0.068367	0.364431	0.534188	0.534189	0.499706	-0.000567	67
	FSFS3	0.591214	0.582864	0.582864	0.595420	0.173878	0.415437	0.587009	0.587024	0.514504	0.029060	31
	MCFS	0.541687	0.539319	0.539319	0.549618	0.080972	0.370332	0.540500	0.540502	0.501116	0.002264	78
RFCM	FSFS1	0.600990	0.588615	0.588615	0.603053	0.189201	0.423222	0.594738	0.594770	0.517557	0.035184	44
	FSFS2	0.600990	0.588615	0.588615	0.603053	0.189201	0.423222	0.594738	0.594770	0.517557	0.035184	51
	FSFS3	0.608140	0.598239	0.598239	0.610687	0.206141	0.431792	0.603149	0.603169	**0.520846**	0.041742	21
	MCFS	0.582609	0.575822	0.575822	0.587786	0.158285	0.407654	0.579196	0.579206	0.511685	0.023417	32
RHFCM	FSFS1	0.521654	0.502582	0.502582	0.541985	0.014955	0.344032	0.511940	0.512029	0.494574	-0.000079	15
	FSFS2	0.534033	0.534272	0.534272	0.534351	0.068305	0.364398	0.534152	0.534152	0.498532	-0.002940	17
	FSFS3	0.591834	0.581573	0.581573	0.595420	0.173103	0.415087	0.586659	0.586681	0.514504	0.029071	10
	MCFS	0.585173	0.570657	0.570657	0.587786	0.155152	0.406296	0.577824	0.577869	0.511685	0.023468	48
RH-IT2FCM	FSFS1	0.522314	0.503064	0.505175	0.546318	0.014985	0.344545	0.512508	0.512598	0.499748	-0.000072	15
	FSFS2	0.534707	0.538045	0.539493	0.536453	0.068585	0.366467	0.536371	0.536373	0.498532	-0.002940	18
	FSFS3	0.595122	0.583898	0.585222	0.600535	0.174498	0.417893	0.589456	0.589483	0.514504	0.029071	9
	MCFS	0.588903	0.572766	0.574950	0.593296	0.156163	0.409168	0.580723	0.580779	0.511685	0.023468	34
EM-SFSCL		0.609144	0.596948	0.596948	0.610687	0.205731	0.431623	0.602984	0.603015	0.520840	0.041754	124
IT2FRH-SFSCL		**0.630748**	**0.628803**	**0.628803**	**0.623853**	**0.259544**	**0.459613**	**0.629774**	**0.629775**	0.512778	**0.276985**	44

performing the grouping, which again proves the interdependency of feature selection and clustering in case of grouping the objects in unsupervised data sets.

The proposed IT2FRH-SFSCL algorithm is finally compared with the existing expectation-maximization based simultaneous feature selection and clustering (EM-SFSCL) algorithm [271]. Tables B.1 - B.5 present the comparative results. Out of 50 cases, in 49 cases, the IT2FRH-SFSCL algorithm provides better results than other clustering algorithms and feature selection-clustering combinations, in terms of different cluster validity indices.

From all the results reported in Tables B.1 - B.5, the conclusions drawn in Section 7.3.3 of Chapter 7 are re-verified.

Execution Time

Moreover, Tables B.1 - B.5 provide the execution time of different algorithms on five data sets. The results reported in Tables B.1 - B.5 establish the fact that the proposed method provides good clustering results in lower or comparable time than most of the existing methods, irrespective of the data sets used. The lower execution time of the proposed technique is achieved due to its low computational complexity with the use of rough hypercuboid approach.

Publications of the Author

International Journal Papers

1. Pradipta Maji and Partha Garai, On Fuzzy-Rough Attribute Selection: Criteria of Max-Dependency, Max-Relevance, Min-Redundancy, and Max-Significance, *Applied Soft Computing*, 13(9):3968–3980, 2013.

2. Pradipta Maji and Partha Garai, Fuzzy-Rough Simultaneous Attribute Selection and Feature Extraction Algorithm, *IEEE Transactions on Cybernetics*, 43(4):1166-1177, 2013.

3. Pradipta Maji and Partha Garai, Simultaneous Feature Selection and Extraction Using Feature Significance, *Fundamenta Informaticae*, 136:405-431, 2015.

4. Pradipta Maji and Partha Garai, IT2 Fuzzy-Rough Sets and Max Relevance-Max Significance Criterion for Attribute Selection, *IEEE Transactions on Cybernetics*, 45(8):1657-1668, 2015.

5. Partha Garai and Pradipta Maji, Generalized Interval Type-2 Fuzzy C-Means Using Rough Hypercuboid Approach, *IEEE Transactions on Cybernetics* (under review).

International Conference Papers

1. Pradipta Maji and Partha Garai, Fuzzy-Rough MRMS Method for Relevant and Significant Attribute Selection, In *Proceedings of the 14th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU '12)*, pages 310–320, Catania, Italy, July 2012.

2. Pradipta Maji and Partha Garai, Simultaneous Feature Selection and Extraction Using Fuzzy-Rough Sets, In *Proceedings of the 2nd International Conference on Soft Computing for Problem Solving (SocProS '12)*, pages 115–123, Rajasthan, India, December 2012.

3. Partha Garai and Pradipta Maji, Identification of Co-Expressed microRNAs Using Rough Hypercuboid Based Interval Type-2 Fuzzy C-Means Algorithm, In *Proceedings of the International Conference on Advanced Computing and Intelligent Engineering (ICACIE '16)*, pages 1–10, Odisha, India, December 2016.

4. Partha Garai and Pradipta Maji, Clustering of microRNAs Using Rough Hypercuboid Based Fuzzy C-Means, In *Proceedings of the 15th International Conference on Information Technology (ICIT '16)*, pages 304–308, Odisha, India, December 2016.

Bibliography

[1] Kent Ridge Bio-medical Data Set Repository. `http://sdmc.lit.org.sg/GEDatasets/Datasets.html/`.

[2] S. S. R. Abidi and K. M. Hoe. Symbolic Exposition of Medical Data-Sets: A Data Mining Workbench To Inductively Derive Data-Defining Symbolic Rules. In *Proceedings of the 15th IEEE Symposium on Computer-Based Medical System (CBMS '02)*, pages 123–128, Maribor, Slovenia, June 2002.

[3] H. Abusamra. A Comparative Study of Feature Selection and Classification Methods for Gene Expression Data of Glioma. *Procedia Computer Science*, 23:5–14, 2013.

[4] C. Affonso, R. J. Sassi, and R. M. Barreiros. Biological Image Classification using Rough-Fuzzy Artificial Neural Network. *Expert Systems with Applications*, 42(24):9482 – 9488, 2015.

[5] D. Agnihotri, K. Verma, and P. Tripathi. Variable Global Feature Selection Scheme for Automatic Classification of Text Documents. *Expert Systems with Applications*, 81:268 – 281, 2017.

[6] A. M. Aibinu, H. B. Salau, N. A. Rahman, M. N. Nwohu, and C. M. Akachukwu. A Novel Clustering Based Genetic Algorithm for Route Optimization. *Engineering Science and Technology, An international Journal*, 19(4):2022 – 2034, 2016.

[7] M. Al-Rajab, J. Lu, and Q. Xu. Examining Applying High Performance Genetic Data Feature Selection and Classification Algorithms for Colon Cancer Diagnosis. *Computer Methods and Programs in Biomedicine*, 146:11 – 24, 2017.

[8] A. Al-Thubaity and A. Al-Subaie. Effect of Word Segmentation on Arabic Text Classification. In *Proceedings of the International Conference on Asian Language Processing (IALP '15)*, pages 127–131, Suzhou, China, October 2015.

[9] M. Alipoor, M. K. Parashkoh, and J. Haddadnia. A Novel Biomarker Discovery Method on Protemic Data for Ovarian Cancer Classification. In *Proceedings of the*

18th Iranian Conference on Electrical Engineering (IECC '10), pages 1–6, Isfahan, Iran, May 2010.

[10] M. Amiri and R. Jensen. Missing Data Imputation Using Fuzzy-Rough Methods. *Neurocomputing*, 205:152–164, 2016.

[11] A. Ammar, Z. Elouedi, and P. Lingras. Semantically Segmented Clustering Based on Possibilistic and Rough Set Theories. *International Journal of Intelligent Systems*, 30(6):676–706, 2015.

[12] S. Anand, K. P. Subbalakshmi, and R. Chandramouli. A Quantitative Model and Analysis of Information Confusion in Social Networks. *IEEE Transactions on Multimedia*, 15(1):207–223, January 2013.

[13] J. R. Anaraki and M. Eftekhari. Improving Fuzzy-Rough Quick Reduct for Feature Selection. In *Proceedings of the 19th Iranian Conference on Electrical Engineering (ICEE '01)*, pages 1501–1506, Tehran, Iran, May 2011.

[14] J. Anderson, A. Pellionisz, and E. Rosenfeld. *Neurocomputing 2: Directions for Research*. Cambridge Mass.: MIT Press, 1990.

[15] V. A. Andrijchuk, I. P. Kuritnyk, M. M. Kasyanchuk, and M. P. Karpinski. Modern Algorithms and Methods of the Person Biometric Identification. In *Proceedings of the IEEE Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS '05)*, pages 403–406, Sofia, Bulgaria, September 2005.

[16] M. Ankerst, M. Breuig, H. P. Kriegel, and J. Sander. OPTICS: Ordering Points to Identity the Clustering Structure. In *Proceedings of the ACM International Conference of Management of Data (SIGMOD '99)*, pages 49–60, Philadelphia, PA, June 1999.

[17] A. Antani, L. R. Long, G. R. Thoma, and R. J. Stanley. Vertebra Shape Classification using MLP for Content-based Image Retrieval. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN '03)*, pages 160–165, Oregon, USA, July 2003.

[18] J. Arias, J. Martnez-Gmez, J. A. Gmez, A. G. S. D. Herrera, and H. Mller. Medical Image Modality Classification using Discrete Bayesian Networks. *Computer Vision and Image Understanding*, 151:61 – 71, 2016.

[19] M. S. Mohd Asaari, S. A. Suandi, and B. A. Rosdi. Geometric Feature Extraction by FTAs for Finger-based Biometrics System. *IET Biometrics*, 6(3):157–164, 2017.

[20] A. Aspin. Tables for Use in Comparisons Whose Accuracy Involves Two Variances. *Biometrika*, 36:245–271, 1949.

[21] W. H. Au and K. C. C. Chan. An Effective Algorithm for Discovering Fuzzy Rules in Relational Databases. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '98)*, pages 1314–1319, Alaska, US, May 1998.

[22] M. Awad, K. Chehdi, and A. Nasri. Multicomponent Image Segmentation Using a Genetic Algorithm and Artificial Neural Network. *IEEE Geoscience and Remote Sensing Letters*, 4(4):571–575, October 2007.

[23] M. Babaee, Y. You, and G. Rigoll. Combined Segmentation, Reconstruction, and Tracking of Multiple Targets in Multi-View Video Sequences. *Computer Vision and Image Understanding*, 154:166 – 181, 2017.

[24] D. L. Bailey, D. W. Townsend, P. E. Valk, and M. N. Maisey. *Positron Emission Tomography: Basic Sciences*. Springer-Verlag, London, 2005.

[25] R. Bajcsy and S. Kovaclc. Multiresolution Elastic Matching. *Computer vision Graphics Image Processing*, 46:1–21, 1989.

[26] R. S. Baker. Educational Data Mining: An Advance for Intelligent Systems in Education. *IEEE Intelligent Systems*, 29(3):78–82, May 2014.

[27] A. Banerjee and P. Maji. Rough Sets for Bias Field Correction in MR Images Using Contraharmonic Mean and Quantitative Index. *IEEE Transactions on Medical Imaging*, 32(11):2140–2151, November 2013.

[28] A. Banerjee and P. Maji. Rough-Probabilistic Clustering and Hidden Markov Random Field Mmodel for Segmentation of HEp-2 Cell and Brain MR Images. *Applied Soft Computing*, 46:558 – 576, 2016.

[29] M. Banerjee, M. K. Kundu, and P. Maji. Content-Based Image Retrieval Using Visually Significant Point Features. *Fuzzy Sets and Systems*, 160(23):3323–3341, 2009.

[30] M. Banerjee, S. Mitra, and H. Banka. Evolutionary Rough Feature Selection in Gene Expression Data. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(4):622–632, July 2007.

[31] N. Barbosa and K. Hirama. Assessment of Software Maintainability Evolution Using C and K Metrics. *IEEE Latin America Transactions*, 11(5):1232–1237, September 2013.

[32] A. Bargiela and W. Pedrycz. *Granular Computing: An Introduction*. Kluwer Academic Publishers, Boston, 2003.

[33] A. Bargiela and W. Pedrycz. Toward a Theory of Granular Computing for Human-Centered Information Processing. *IEEE Transactions on Fuzzy Systems*, 16(2):320–330, 2008.

[34] R. Battiti. Using Mutual Information for Selecting Features in Supervised Neural Net Learning. *IEEE Transactions on Neural Network*, 5(4):537–550, 1994.

[35] I. Bayraktaroglu and A. Orailoglu. The Construction of Optimal Deterministic Partitionings in Scan-Based BIST Fault Diagnosis: Mathematical Foundations and Cost-Effective Implementations. *IEEE Transactions on Computers*, 54(1):61–75, January 2005.

[36] T. Beaubouef, F. E. Petry, and G. Arora. Information-Theoretic Measures of Uncertainity for Rough Sets and Rough Relational Databases. *Information Sciences*, 109(1-4):185–195, 1998.

[37] P. W. Becker. *Recognition of Patterns*. Vienna: Springer-Verlag, 1978.

[38] A. H. Beg, M. Z. Islam, and V. Estivill-Castro. Genetic Algorithm with Healthy Population and Multiple Streams Sharing information for Clustering. *Knowledge-Based Systems*, 114:61 – 78, 2016.

[39] I. Beheshti, H. Demirel, and H. Matsuda. Classification of Alzheimer's Disease and Prediction of Mild Cognitive Impairment-To-Alzheimer's Conversion from Structural Magnetic Resource Imaging using Feature Ranking and A Genetic Algorithm. *Computers in Biology and Medicine*, 83:109 – 119, 2017.

[40] E. J. Bekkers and C. A. Taylor. Multiscale Vascular Surface Model Generation From Medical Imaging Data Using Hierarchical Features. *IEEE Transactions on Medical Imaging*, 27(3):331–341, March 2008.

[41] N. Belacel, M. Cuperlovic-Culf, M. Laflamme, and R. Ouellette. Fuzzy J-Means and VNS Methods for Clustering Genes from Microarray Data. *Bioinformatics*, 20(11):1690–1701, 2004.

[42] M. Belhor and F. Jemili. Intrusion Detection based on Genetic Fuzzy Classification System. In *Proceedings of the 13th International Conference of Computer Systems and Applications (AICCSA '16)*, pages 1–8, November 2016.

[43] A. J. Bell and T. J. Sejnowski. An Information-Maximization Approach to Blind Separation and Blind Deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

[44] R. E. Bellman, R. E. Kalaba, and L. A. Zadeh. Abstraction and Pattern Classification. *Journal of Mathematical Analysis and Applications*, 13:1–7, 1966.

[45] A. Ben-Dor, R. Shamir, and Z. Yakhini. Clustering Gene Expression Patterns. *Journal of Computational Biology*, 6(3-4):281–297, 1999.

[46] T. V. Benzel, E. O'Brien, R. Rodriguez, W. Arbaugh, and J. Sebes. Crossing the Great Divide: From Research to Market. *IEEE Security Privacy*, 11(2):42–46, March 2013.

[47] H.-G. Beyer and H.-P. Schwefel. Evolution Strategies – A Comprehensive Introduction. *Natural Computing*, 1(1):3–52, May 2002.

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

[49] J. C. Bezdek, L. O. Hall, and L. P. Clarke. Review of MR Image Segmentation Techniques Using Pattern Recognition. *Medical Physics*, 20(4):1033–1048, 1993.

[50] J. C. Bezdek and N. R. Pal. Some New Indexes for Cluster Validity. *IEEE Transactions on System, Man, and Cybernetics, Part B: Cybernetics*, 28(3):301–315, 1988.

[51] J. C. Bezdek and S. K. Pal. *Fuzzy Models for Pattern Recognition: Methods that Search for Structures in Data.* IEEE Press, New York, 1992.

[52] S. K. Bhatia and J. S. Deogun. Conceptual Clustering in Information Retrieval. *IEEE Transactions on Systems, Man and Cybernatics, Part B: Cybernatics*, 28(3):427–436, 1998.

[53] S. N. B. Bhushan and A. Danti. Classification of Text Documents based on Score Level Fusion Approach. *Pattern Recognition Letters*, 94:118–126, 2017.

[54] R. Binisha. Ontology based Text Clustering using the Dissimilarity Measure. In *Proceedings of the International Conference on Communication and Computational Intelligence (INCOCCI '10)*, pages 476–480, Erode, India, December 2010.

[55] J. Bins and B. A. Draper. Feature Selection from Huge Feature Sets. In *Proceedings of the 8th International Conference on Computer Vision (ICCV '01)*, pages 159–165, Vancouver, Canada, July 2001.

[56] T. Blickle and L. Thiele. A Comparison of Selection Schemes Used in Evolutionary Algorithms. *Evolutionary Computation*, 4(4):361–394, 1996.

[57] A. L. Blum and P. Langley. Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence*, 97:245–271, 1997.

[58] I. Borg and P. Groenen. *Modern Multidimensional Scaling*. Springer-Verlag, Berlin, 1997.

[59] P. Bosc, O. Pivert, and L. Ughetto. Database Mining for the Discovery of Extended Functional Dependencies. In *Proceedings of the 18th International Conference of the North American Fuzzy Information Processing Society (NAFIPS '99)*, pages 580–584, New York, USA, June 1999.

[60] E. I. Boyle, S. Weng, J. Gollub, H. Jin, D. Botstein, J. M. Cherry, and G. Sherlock. GO::Term Finder Open Source Software for Accessing Gene Ontology Information and Finding Significantly Enriched Gene Ontology Terms Associated with a List of Genes. *Bioinformatics*, 20:3710–3715, 2004.

[61] M. Bressan and J. Vitria. On the Selection and Classification of Independent Features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1312–1317, 2003.

[62] D. E. Brown, L. E. Gunderson, and M. H. Evans. Interactive Analysis of Computer Crimes. *Computer*, 33(8):69–77, August 2000.

[63] R. Brunelli. *Template Matching Techniques in Computer Vision: Theory and Practice*. Wiley Publishing, 2009.

[64] H. Bunke and A. Kandel. *Neuro-Fuzzy Pattern Recognition*. World Scientific, Singapore, 2001.

[65] H. Bustince and P. Burillo. Correlation of Interval-Valued Intuitionistic Fuzzy Sets. *Fuzzy Sets and Systems*, 74(2):237–244, 1995.

[66] D. Cai, C. Zhang, and X. He. Unsupervised Feature Selection for Multi-cluster Data. In *Proceedings of the 16th ACM International Conference on Knowledge Discovery and Data Mining (KDD '10)*, pages 333–342, Washington DC, USA, July 2010.

[67] S. Camiz and V. D. Pillar. Comparison of Single and Complete Linkage Clustering with the Hierarchical Factor Classification of Variables. *Community Ecology*, 8(1):25–30, 2007.

[68] J. Cardoso. Blind Signal Separation: Statistical Principles. *Proceedings of IEEE*, 86:2009–2025, 1998.

[69] C. Carpineto and G. Romano. A Lattice Conceptual Clustering System and Its Application to Browsing Retrieval. *Machine Learning*, 24(2):95–122, 1996.

[70] H. Causton, J. Quackenbush, and A. Brazma. *Microarray Gene Expression Data Analysis: A Beginner's Guide*. Wiley-Blackwell, 2003.

[71] D. Chakrabarti, R. Kumar, and A. Tomkins. Evolutionary clustering. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '06)*, pages 554–560, Philadelphia, PA, USA, August 2006.

[72] B. Chakraborty, K. Chakma, and A. Mukherjee. A Density-Based Clustering Algorithm And Experiments on Student Dataset With Noises Using Rough Set Theory. In *Proceedings of the IEEE International Conference on Engineering And Technology (ICETECH '16)*, pages 431–436, Coimbatore, India, March 2016.

[73] A. Charoenrook and M. Soma. A Fault Diagnosis Technique for Flash ADC's. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 43(6):445–457, June 1996.

[74] A. Chatterjee, A. L. Leichter, V. Fan, P. Tsai, R. V. Purcell, M. J. Sullivan, and M. R. Eccles. A Cross Comparison of Technologies for the Detection of microRNAs in Clinical FFPE Samples of Hepatoblastoma Patients. *Scientific Reports*, 5, 2015.

[75] M. H. Chehreghani, H. Abolhassani, and M. H. Chehreghani. Improving Density-based Methods for Hierarchical Clustering of Web Pages. *Data and Knowledge Engineering*, 67(1):30–50, 2008.

[76] D. Chen, L. Zhang, S. Zhao, Q. Hu, and P. Zhu. A Novel Algorithm for Finding Reducts With Fuzzy Rough Sets. *IEEE Transactions on Fuzzy Systems*, 20(2):385–389, 2012.

[77] S. Y. Chen, Y. Wang, C. C. Wu, C. Liu, and C. I. Chang. Real-Time Causal Processing of Anomaly Detection for Hyperspectral Imagery. *IEEE Transactions on Aerospace and Electronic Systems*, 50(2):1511–1534, April 2014.

[78] W.-H. Chen, C.-W. Liu, and M.-S. Tsai. On-line Fault Diagnosis of Distribution Substations using Hybrid Cause-Effect Network and Fuzzy Rule-Based Method. *IEEE Transactions on Power Delivery*, 15(2):710–717, April 2000.

[79] Y. Chen, Z. Zhang, J. Zheng, Y. Ma, and Y. Xue. Gene Selection for Tumor Classification using Neighborhood Rough Sets and Entropy Measures. *Journal of Biomedical Informatics*, 67:59 – 68, 2017.

[80] D.-A. Chiang, L. R. Chow, and Y.-F. Wang. Mining Time Series Data by a Fuzzy Linguistic Summary System. *Fuzzy Sets and Systems*, 112:419–432, 2000.

[81] J.-H. Chiang and S.-H. Ho. A Combination of Rough-Based Feature Selection and RBF Neural Network for Classification Using Gene Expression Data. *IEEE Transactions on NanoBioscience*, 7(1):91–99, 2008.

[82] C.-H. Chou, S.-C. Hsieh, and C.-J. Qiu. Hybrid Genetic Algorithm and Fuzzy Clustering for Bankruptcy Prediction. *Applied Soft Computing*, 56:298 – 316, 2017.

[83] A. Chouchoulas and Q. Shen. Rough Set-Aided Keyword Reduction for Text Categorisation. *Applied Artificial Intelligence*, 15(9):843–873, 2001.

[84] Feng Chu, Wei Xie, and Lipo Wang. Gene Selection and Cancer Classification using a Fuzzy Neural Network. In *Processing of the IEEE Annual Meeting of the Fuzzy Information (NAFIPS '04)*, volume 2, pages 555–559, Alberta, Canada, June 2004.

[85] T. F. Cox and M. A. A. Cox. *Multidimensional Scaling, Second Edition.* Chapman and Hall/CRC, 2000.

[86] Y. Le. Cun, B. Boser, J. S. Denker, D. Henderson, R. E. Horward, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4):541–551, 1989.

[87] R. Damaeviius, R. Maskelinas, A. Venkauskas, and M. Woniak. Smartphone User Identity Verification Using Gait Characteristics. *Symmetry*, 8(10), 2016.

[88] A. K. Das, S. Das, and A. Ghosh. Ensemble Feature Selection using Bi-Objective Genetic Algorithm. *Knowledge-Based Systems*, 123:116 – 127, 2017.

[89] S. K. Das. Feature Selection with a Linear Dependence Measure. *IEEE Transactions on Computers*, 20(9):1106–1109, 1971.

[90] K. S. Dash, N.B. Puhan, and G. Panda. BESAC: Binary External Symmetry Axis Constellation for Unconstrained Handwritten Character Recognition. *Pattern Recognition Letters*, 83, Part 3:413 – 422, 2016.

[91] M. Dash and H. Liu. Feature Selection for Classification. *Intelligent Data Analysis*, 1(1–4):131–156, 1997.

[92] M. Dash and H. Liu. Unsupervised Feature Selection. In *Proceedings of the Pacific Asia Conference on Knowledge Discovery and Data Mining (PAKDD '00)*, pages 110–121, Kyoto, Japan, April 2000.

[93] M. Dash and H. Liu. Consistency Based Search in Feature Selection. *Artificial Intelligence*, 151(1-2):155–176, 2003.

[94] D. L. Davies and D. W. Bouldin. A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(2):224–227, 1979.

[95] W. H. E. Day and H. Edelsbrunner. Efficient Algorithms for Agglomerative Hierarchical Clustering Methods. *Journal of Classification*, 1(1):7–24, 1984.

[96] S. K. De. A Rough Set Theoretic Approach to Clustering. *Fundamenta Informaticae*, 62(3-4):409–417, 2004.

[97] K. Deb and M. Koksalan. Special Issue on Preference-Based Multiobjective Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation*, 13(3):674–674, June 2009.

[98] D. Dembele and P. Kastner. Fuzzy C-Means Method for Clustering Microarray Data. *Bioinformatics*, 19(8):973–980, 2003.

[99] S. Derhami and A. E. Smith. A Technical Note on the Paper Hga: Hybrid Genetic Algorithm in Fuzzy Rule-Based Classification Systems for High-Dimensional Problems. *Applied Soft Computing*, 41:91 – 93, 2016.

[100] N. Dess, E. Pascariello, and B. Pes. A Comparative Analysis of Biomarker Selection Techniques. *BioMed Research International*, 2013.

[101] P. A. Devijver and J. Kittler. *Pattern Recognition: A Statistical Approach*. Prentice Hall, Englewood Cliffs, 1982.

[102] P. A. Devijver and J. Kittler. *Pattern Recognition: Theory and Applications*. Berlin: Springer-Verlag, 1987.

[103] L. Devroye, L. Gyorfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer-Verlag, Berlin, 1996.

[104] L. R. Dice. Measures of the Amount of Ecologic Association Between Species. *Ecology*, 26(3):297–302, 1945.

[105] C. Ding and H. Peng. Minimum Redundancy Feature Selection from Microarray Gene Expression Data. *Journal of Bioinformatics and Computational Biology*, 3(2):185–205, 2005.

[106] S. Ding, J. Chen, X. Xu, and J. Li. Rough Neural Networks: A Review. *Journal of Computational Information Systems*, 7(7):2338–2346, 2011.

[107] Y. Ding and X. Fu. Kernel-Based Fuzzy C-Means Clustering Algorithm Based on Genetic Algorithm. *Neurocomputing*, 188:233 – 238, 2016.

[108] E. Domany. Cluster Analysis of Gene Expression Data. *Journal of Statistical Physics*, 110(3-6):1117–1139, 2003.

[109] Y. K. Dubey, M. M. Mushrif, and K. Mitra. Segmentation of Brain MR Images using Rough Set Based Intuitionistic Fuzzy Clustering. *Biocybernetics and Biomedical Engineering*, 36(2):413 – 426, 2016.

[110] D. Dubois and H. Prade. Rough Fuzzy Sets and Fuzzy Rough Sets. *International Journal of General Systems*, 17:191–209, 1990.

[111] D. Dubois and H. Prade. Putting Fuzzy Sets and Rough Sets Together. In *Intelligent Decision Support: Handbook of Applications and Advances of Rough Sets Theory*, pages 203–232. Kluwer, 1992.

[112] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification and Scene Analysis*. John Wiley and Sons, New York, 1999.

[113] J. C. Dunn. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernatics*, 3:32–57, 1973.

[114] S. Dutta and A. Skowron. Generalized Quantifiers In The Context of Rough Set Semantics. *Fundamenta Informaticae*, 142(1-4):213–236, 2015.

[115] B. Efron and R. Tibshirani. Estimating the Error Rate of A Prediction Rule: Improvement on Cross-Validation. *American Statistical Association*, 78(382):316–331, 1983.

[116] B. Efron and R. Tibshirani. Improvements on Cross-Validation: The .632+ Bootstrap Method. *Journal of the American Statistical Association*, 92(438):548–560, 1997.

[117] M. B. Eisen, P. T. Spellman, O. Patrick, and D. Botstein. Cluster Analysis and Display of Genome-Wide Expression Patterns. *Proceedings of the National Academy of Sciences, USA*, 95(25):14863–14868, 1998.

[118] I. Elafi, M. Jedra, and N. Zahid. Unsupervised Detection and Tracking of Moving Objects for Video Surveillance Applications. *Pattern Recognition Letters*, 84:70 – 77, 2016.

[119] T. Elguebaly and N. Bouguila. Simultaneous High-Dimensional Clustering and Feature Selection using Asymmetric Gaussian Mixture Models. *Image and Vision Computing*, 34:27–41, 2015.

[120] S. Ensafi, S. Lu, A. A. Kassim, and C. L. Tan. Automatic CAD System for HEp-2 Cell Image Classification. In *Proceedings of the 22nd International Conference on Pattern Recognition (ICPR '14)*, pages 3321–3326, Stockholm, Sweden, August 2014.

[121] S. Erdweg, S. Fehrenbach, and K. Ostermann. Evolution of Software Systems with Extensible Languages and DSLs. *IEEE Software*, 31(5):68–75, September 2014.

[122] D. Y. Eroglu and K. Kilic. A Novel Hybrid Genetic Local Search Algorithm for Feature Selection and Weighting with an Application in Strategic Decision Making in innovation Management. *information Sciences*, 405:18 – 32, 2017.

[123] J. 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 the 2nd International Conference on Knowledge Discovery and Data Mining (KDD '96)*, pages 226–231, Portland, OR, August 1996.

[124] M. Ester, H. P. Kriegel, and X. Xu. Knowledge Discovery in Large Spatial Databaes: Focusing Techniques for Efficient Class Identification. In *Proceedings of the International Symposium on Large Spatial Databases (SSTD '95)*, pages 67–82, Portland, ME, August 1995.

[125] A. Farasat, G. Gross, R. Nagi, and A. G. Nikolaev. Social Network Analysis With Data Fusion. *IEEE Transactions on Computational Social Systems*, 3(2):88–99, June 2016.

[126] H. M. Fayek, M. Lech, and L. Cavedon. Evaluating Deep Learning Architectures for Speech Emotion Recognition. *Neural Networks*, 92:60 – 68, 2017.

[127] D. Fisher. Improving Inference through Conceptual Clustering. In *Proceedings of the National Conference of Artificial Intelligence (AAAI '87)*, pages 461–465, Seattle, WA, July 1987.

[128] D. B. Fogel and L. J. Fogel. *An Introduction to Evolutionary Programming*, pages 21–33. Springer Berlin Heidelberg, Berlin, Heidelberg, 1996.

[129] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley and Sons, New York, 1966.

[130] X. Font-Aragones, M. Faundez-Zanuy, and J. Mekyska. Thermal Hand Image Segmentation for Biometric Recognition. *IEEE Aerospace and Electronic Systems Magazine*, 28(6):4–14, June 2013.

[131] G. Forman. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. *The Journal of Machine Learning Research*, 3:1289–1305, 2003.

[132] C. Fraley and A. E. Raftery. How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. *The Computer Journal*, 41(8):578–588, 1998.

[133] C. Fraley and A. E. Raftery. Model-Based Clustering, Discriminant Analysis, and Density Estimation. *Journal of the American Statistical Association*, 97(458):611–632, 2002.

[134] A. Frank and A. Asuncion. UCI Machine Learning Repository. `http://archive.ics.uci.edu/ml`, 2010.

[135] M. M. Fraz, W. Jahangir, S. Zahid, M. M. Hamayun, and S. A. Barman. Multiscale Segmentation of Exudates in Retinal Images using Contextual Cues and Ensemble Classification. *Biomedical Signal Processing and Control*, 35:50 – 62, 2017.

[136] J. H. Friedman. Exploratory Projection Pursuit. *Journal of the American Statistical Association*, 82(397):249–266, 1987.

[137] H. Frigui. Adaptive Image Retrieval Using the Fuzzy Integral. In *Proceedings of the 18th International Conference of the North American Fuzzy Information Processing Society (NAFIPS '99)*, pages 575–579, New York, USA, 1999.

[138] H. Frigui and R. Krishnapuram. A Robust Competitive Clustering Algorithm with Applications in Computer Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):450–465, 1999.

[139] K. S. Fu. *Syntactic Pattern Recognition and Applications*. Academic Press, London, 1982.

[140] L. Fu. An Expert Network for DNA Sequence Analysis. *IEEE Intelligent Systems and their Applications*, 14(1):65–71, January 1999.

[141] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, New York, 1990.

[142] K. Fukushima, S. Miyako, and T. Ito. Neocognitron: A Neural Network Model for a Mechanism of Visual Pattern Recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 13:826–834, 1983.

[143] F. Galland, N. Bertaux, and P. Refregier. Minimum Description Length Synthetic Aperture Radar Image Segmentation. *IEEE Transactions on Image Processing*, 12(9):995–1006, September 2003.

[144] Y. Gao, X. Qi, and F. Palmieri. Comments on "Theoretical Analysis of Evolutionary Algorithms with an Infinite Population Size in Continuous Space. I. Basic Properties of Selection and Mutation". *IEEE Transactions on Neural Networks*, 9(2):341–343, March 1998.

[145] P. Garai and P. Maji. Generalized Interval Type-2 Fuzzy C-Means Using Rough Hypercuboid Approach. *IEEE Transactions on Cybernetics, Under Review*, 99(1):1–16.

[146] P. Garai and P. Maji. Clustering of microRNAs using Rough Hypercuboid based Fuzzy C-Means. In *Proceedings of the 15th International Conference on Information Technology (ICIT '16)*, pages 304–308, Odisha, India, December 2016.

[147] P. Garai and P. Maji. Identification of Co-Expressed microRNAs Using Rough Hypercuboid Based Interval Type-2 Fuzzy C-Means Algorithm. In *Proceedings of the International Conference on Advanced Computing and Intelligent Engineering (ICACIE '16)*, pages 1–10, Odisha, India, December 2016.

[148] A. Garcia-Piquer, J. Bacardit, A. Fornells, and E. Golobardes. Scaling-Up Multiobjective Evolutionary Clustering Algorithms using Stratification. *Pattern Recognition Letters*, 93:69 – 77, 2017.

[149] R. M. Garimella, M. Gabbouj, and I. Ahmad. Image Retrieval: Information and Rough Set Theories. *Procedia Computer Science*, 54:631 – 637, 2015.

[150] I. Gath and A. Geva. Unsupervised Optimal Fuzzy Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):773–780, 1989.

[151] A. Ghosh, N. R. Pal, and S. K. Pal. Image Segmentation Using Neural Networks. *Biological Cybernetics*, 66(2):151–158, 1991.

[152] R. Gilad-Bachrach, A. Navot, and N. Tishby. Margin Based Feature Selection: Theory and Algorithms. In *Proceedings of the 21st International Conference on Machine Learning (ICML '04)*, Alberta, Canada, July 2004.

[153] M. K. Goel. An Overview of Brain Computer Interface. In *Proceedings of the Recent and Emerging trends in Computer and Computational Sciences (RETCOMP '15)*, pages 10–17, Bangalore, India, January 2015.

[154] J. A. Goguen. L-fuzzy Sets. *Journal of Mathematical Analysis and Applications*, 18(1):145–174, 1967.

[155] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989.

[156] A. Gomes, C. H. Antunes, and A. G. Martins. A Multiple Objective Approach to Direct Load Control Using an Interactive Evolutionary Algorithm. *IEEE Transactions on Power Systems*, 22(3):1004–1011, August 2007.

[157] J. B. Gomez-Mendoza, F. Prieto, and H. Redarce. Automatic Lip-Contour Extraction and Mouth-Structure Segmentation in Images. *Computing in Science Engineering*, 13(3):22–30, May 2011.

[158] L. B. Goncalves, M. M. B. R. Vellasco, M. A. C. Pacheco, and F. J. de Souza. Inverted Hierarchical Neuro-Fuzzy BSP System: A Novel Neuro-Fuzzy Model for Pattern Classification and Rule Extraction in Databases. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 36(2):236–248, 2006.

[159] P. Gong and E. J. Perkins. Earthworm Toxicogenomics: A Renewed Genome-Wide Quest for Novel Biomarkers and Mechanistic Insights. *Applied Soil Ecology*, 104:12 – 24, 2016.

[160] C. Gottschlich, B. Tams, and S. Huckemann. Perfect Fingerprint Orientation Fields by Locally Adaptive Global Models. *IET Biometrics*, 6(3):183–190, 2017.

[161] U. Grenander. *General Pattern Theory.* Oxford University Press, 1993.

[162] M. Griffiths and H. Sundaram. Drug Design and Testing: Profiling of Antiproliferative Agents for Cancer Therapy using a Cell-Based Methyl-[3H]-Thymidine Incorporation Assay. *Methods in Molecular Biology*, 731:451–465, 2011.

[163] F. Grisoni, V. Consonni, M. Vighi, S. Villa, and R. Todeschini. Investigating The Mechanisms of Bioconcentration Through QSAR Classification Trees. *Environment International*, 88:198 – 205, 2016.

[164] R. Gruter, O. Egger, J. M. Vesin, and M. Kunt. Rank-Order Polynomial Subband Decomposition for Medical Image Compression. *IEEE Transactions on Medical Imaging*, 19(10):1044–1052, October 2000.

[165] S. Guha, R. Rastogi, and K. Shim. Rock: A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5):345 – 366, 2000.

[166] V. Guillemot, C. Philippe, A. Tenenhaus, J. Rollin, X. Gidrol, and V. Frouin. Grouping Levels of Exposure with Same Observable Effects Before Class Prediction in Toxicogenomics. In *Proceedings of the 2008 International Conference on Biocomputation, Bioinformatics, and Biomedical Technologies (BIOTECHNO '08)*, pages 164–169, Bucharest, Romania, June 2008.

[167] M. Gupta, P. Kumar, and B. Bhasker. Clustering of Users on Microblogging Social Media: A Rough Set Based Approach. In *Proceedings of the International Conference On Data Science And Engineering (ICDSE '16)*, pages 1–6, Kerala, India, August 2016.

[168] D. E. Gustafson and W. C. Kessel. Fuzzy Clustering with a Fuzzy Covariance Matrix. In *Proceedings of the 1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, pages 761–766, San Diego, USA, January 1978.

[169] I. Guyon and A. Elisseeff. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[170] L. G. Hafemann, R. Sabourin, and L. S. Oliveira. Learning Features for Offline Handwritten Signature Verification using Deep Convolutional Neural Networks. *Pattern Recognition*, 70:163 – 176, 2017.

[171] H. Hagras. A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots. *IEEE Transactions on Fuzzy Systems*, 12(4):524–539, 2004.

[172] H. Hakli and H. Uguz. A Novel Approach for Automated Land Partitioning using Genetic Algorithm. *Expert Systems with Applications*, 82:10 – 18, 2017.

[173] M. A. Hall. Correlation Based Feature Selection for Discrete and Numerical Class Machine Learning. In *Proceedings of the 17th International Conference on Machine Learning (ICML '00)*, Standford, USA, June 2000.

[174] M. Hamadache, D. Lee, and K. C. Veluvolu. Rotor Speed-Based Bearing Fault Diagnosis (RSB-BFD) Under Variable Speed and Constant Load. *IEEE Transactions on Industrial Electronics*, 62(10):6486–6495, October 2015.

[175] J. Han and M. Kamber. *Data Mining, Concepts and Techniques*. Morgan Kaufmann, 1991.

[176] R. Han and J. Liu. Spectral Clustering and Genetic Algorithm for Design of District Metered Areas in Water Distribution Systems. *Procedia Engineering*, 186:152 – 159, 2017.

[177] R. Haralick and L. Shapiro. Survey: Image Segmentation Techniques. *Computer Vision, Graphics and Image Processing*, 29:100–132, 1985.

[178] M. Harman, B. Korel, and P. K. Linos. Guest Editorial: Special Issue on Software Maintenance and Evolution. *IEEE Transactions on Software Engineering*, 31(10):801–803, October 2005.

[179] W. E. Hart. Evolutionary Pattern Search Algorithms for Unconstrained and Linearly Constrained Optimization. *IEEE Transactions on Evolutionary Computation*, 5(4):388–397, August 2001.

[180] E. Hartuv and R. Shamir. A Clustering Algorithm Based on Graph Connectivity. *Information Processing Letters*, 76(4-6):175–181, 2000.

[181] B. Harvey and S. Y. Ji. Cloud-Scale Genomic Signals Processing Classification Analysis for Gene Expression Microarray Data. In *Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '14)*, pages 1843–1846, Chicago, USA, August 2014.

[182] A. Hassan, M. U. H. Zaka, D. Kouvatsos, and Y. Peng. Integrated Analysis of Gene Expression Data for Colon Cancer Biomarker Discovery. In *Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI '15)*, pages 536–541, Capetown, South Africa, December 2015.

[183] A. E. Hassanien, A. Abraham, J. F. Peters, and G. Schaefer. An Overview of Rough-Hybrid Approaches in Image Processing. In *Proceedings of the 16th International Conference on IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '08)*, pages 2135–2142, Hong Kong, China, June 2008.

[184] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, New Jersey, USA, 2nd edition, 1998.

[185] Q. He, T. Li, F. Zhuang, and Z. Shi. Frequent Term based Peer-to-peer Text Clustering. In *Proceedings of the 3rd International Symposium on Knowledge Acquisition and Modeling (KAM '10)*, pages 352–355, Wuhan, China, October 2010.

[186] Z. He, S. Lin, Y. Deng, X. Li, and Q. Qian. A Rough Membership Neural Network Approach for Fault Classification in Transmission Lines. *International Journal of Electrical Power and Energy Systems*, 61:429 – 439, 2014.

[187] R. Hecht-Nielsen. *Neurocomputing*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.

[188] R. P. Heydorn. Redundancy in Feature Extraction. *IEEE Transactions on Computers*, 20(9):1051–1054, 1971.

[189] A. Hinneburg and D. Keim. An Efficient Approach to Clustering Large Multimedia Databases with Noise. In *Proceedings of the 4th ACM Conference on Knowledge Discovery and Data Mining (KDD '98)*, pages 58–65, New York, USA, August 1998.

[190] A. Hinneburg and D. A. Keim. Optimal Grid-Clustering: Towards Breaking the Curse of Dimensionality in High-Dimensional Clustering. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB '99)*, pages 506–517, Edinburgh, Scotland, 1999.

[191] A. E. Hoerl and R. W. Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 42(1):80–86, 2000.

[192] U. Hohle. Quotients with Respect to Similarity Relations. *Fuzzy Sets and Systems*, 27(1):31–44, 1988.

[193] S. C. H. Hoi, J. Wang, P. Zhao, and R. Jin. Online Feature Selection for Mining Big Data. In *Proceedings of the 1st International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications (BigMine '12)*, pages 93–100, Beijing, China, 2012.

[194] J. H. Holland. *Adaptation in Natural and Artificial Systems.* Ann Arbor: The University of Michigan Press, 1975.

[195] K. Hu, Y. Lu, and C. Shi. Feature ranking in rough sets. *AI Communications - Artificial Intelligence Advances in China*, 16(1):41–50, 2003.

[196] Q. Hu, S. Chakhar, S. Siraj, and A. Labib. Spare Parts Classification in Industrial Manufacturing using The Dominance-Based Rough Set Approach. *European Journal of Operational Research*, pages 1 – 14, 2017.

[197] Q. Hu, W. Pan, S. An, P. Ma, and J. Wei. An Efficient Gene Selection Technique for Cancer Recognition Based on Neighborhood Mutual Information. *International Journal of Machine Learning and Cybernetics*, 1(1-4):63–74, 2010.

[198] Q. Hu, D. Yu, J. Liu, and C. Wu. Neighborhood Rough Set Based Heterogeneous Feature Subset Selection. *Information Sciences*, 178:3577–3594, 2008.

[199] Q. Hu, D. Yu, Z. Xie, and J. Liu. Fuzzy Probabilistic Approximation Spaces and Their Information Measures. *IEEE Transactions on Fuzzy Systems*, 14(2):191–201, 2007.

[200] Y. Hu, N. Yu, Z. Li, and M. Li. Image Search Result Clustering and Re-Ranking via Partial Grouping. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '07)*, pages 603–606, Beijing, China, July 2007.

[201] Y.-C. Hu. Tolerance Rough Sets for Pattern Classification using Multiple Grey Single-Layer Perceptrons. *Neurocomputing*, 179:144 – 151, 2016.

[202] D. Huang and T. W. S. Chow. Effective Feature Selection Scheme Using Mutual Information. *Neurocomputing*, 63:325–343, 2004.

[203] D. Huang, T. W. S. Chow, E. W. M. Ma, and Jinyan Li. Efficient Selection of Discriminative Genes from Microarray Gene Expression Data for Cancer Diagnosis. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 52(9):1909–1918, September 2005.

[204] Z. Huang. Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values. *Data Mining and Knowledge Discovery*, 2:283–304, 1998.

[205] L. Hubert and P. Arabie. Comparing Partitions. *Journal of Classification*, 2(1):193–218, 1985.

[206] C. F. Hwang and C. H. Rhee. Uncertain Fuzzy Clustering: Interval Type-2 Fuzzy Approach to C-Means. *IEEE Transactions on Fuzzy Systems*, 15(1):107–120, 2007.

[207] A. Hyvarinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley & Sons, 2001.

[208] A. Hyvärinen and E. Oja. Independent Component Analysis: Algorithms and Applications. *Neural Networks*, 13(4-5):411–430, May 2000.

[209] A. Hyvarlnen and R. Oja. A Fast Fixed-Point Algorithm for Independent Component Analysis. *Neural Computation*, 9(7):1483–1492, October 1997.

[210] S. Ioffe. Probabilistic Linear Discriminant Analysis. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part IV (ECCV '06)*, pages 531–542, Berlin, Heidelberg, 2006.

[211] R. Isermann. On Fuzzy Logic Applications for Automatic Control, Supervision, and Fault Diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 28(2):221–235, March 1998.

[212] S. M. Islam, M. Banerjee, S. Bhattacharyya, and S. Chakraborty. Content-Based Image Retrieval Based on Multiple Extended Fuzzy-Rough Framework. *Applied Soft Computing*, 57:102 – 117, 2017.

[213] D.E. Nelson J. Starzyk and K. Sturtz. Reduct Generation in Information Systems. *Bulletin of International Rough Set Society*, 1(1–2):19–22, 1999.

[214] P. Jaccard. Etude Comparative de la Distribution Florale dans une Portion des Alpes et des Jura. *Bulletin de la Societe Vaudoise des Sciences Naturelles*, 37:547–579, 1901.

[215] P. Jaganathan, K. Thangavel, A. Pethalakshmi, and M. Karnan. Classification Rule Discovery with Ant Colony Optimization And Improved Quick Reduct Algorithm. *IAENG International Journal of Computer Science*, 33(1):50–55, 2007.

[216] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Englewood Cliffs, NJ: Prentice Hall, 1988.

[217] A. K. Jain, J. Mao, and K. M. Mohiuddin. Artificial Neural Networks: A Tutorial. *Computer*, pages 31–44, 1996.

[218] A. K. Jain, M. N. Murty, and P. J. Flynn. Data Clustering: A Review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[219] R. Jensen. *Combining Rough and Fuzzy Sets for Feature Selection*. PhD thesis, University of Edinburgh, UK, 2005.

[220] R. Jensen and Q. Shen. Semantics-Preserving Dimensionality Reduction: Rough and Fuzzy-Rough-Based Approach. *IEEE Transactions on Knowledge and Data Engineering*, 16(12):1457–1471, 2004.

[221] R. Jensen and Q. Shen. Fuzzy-Rough Sets Assisted Attribute Selection. *IEEE Transactions on Fuzzy Systems*, 15:73–89, 2007.

[222] R. Jensen and Q. Shen. *Computational Intelligence and Feature Selection: Rough and Fuzzy Approaches*. Wiley-IEEE Press, 2008.

[223] R. Jensen and Q. Shen. New Approaches to Fuzzy-Rough Feature Selection. *IEEE Transactions on Fuzzy Systems*, 17(4):824–838, 2009.

[224] G. Jeon and J. Jeong. *Rough Sets-Based Image Processing for Deinterlacing*, pages 227–239. 2008.

[225] W. Jia, D. Zhao, and L. Ding. An Optimized RBF Neural Network Algorithm Based on Partial Least Squares and Genetic Algorithm for Classification of Small Sample. *Applied Soft Computing*, 48:373 – 384, 2016.

[226] D. Jiang, C. Tang, and A. Zhang. Cluster Analysis for Gene Expression Data: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 16(11):1370–1386, 2004.

[227] J. Jiang, D. Yang, and H. Wei. Image Segmentation Based on Rough Set Theory and Neural Networks. In *Proceedings of the 5th International Conference on Visual Information Engineering (VIE '08)*, pages 361–365. IET, UK, July 2008.

[228] S. Jiang, K. Chin, L. Wang, G. Qu, and K. L. Tsui. Modified Genetic Algorithm-Based Feature Selection Combined with Pre-Trained Deep Neural Network for Demand Forecasting in Outpatient Department. *Expert Systems With Applications*, 82:216 – 230, 2017.

[229] X.-R. Jiang and L. Gruenwald. Microarray Gene Expression Data Association Rules Mining based on JG-Tree. In *Proceedings of the 14th International Workshop on Database and Expert Systems Applications (DeXA '03)*, pages 27–31, Prague, Czech Republic, September 2003.

[230] L. Jie, W. Liu, Z. Sun, and S. Teng. Hybrid Fuzzy Clustering Methods based on Improved Self-Adaptive Cellular Genetic Algorithm and Optimal-Selection-based Fuzzy C-Means. *Neurocomputing*, 249:140 – 156, 2017.

[231] G. H. John, R. Kohavi, and K. Pfleger. Irrelevant Features and the Subset Selection Problem. In *Proceedings of the 11th International Conference on Machine Learning (ICML '94)*, pages 121–129. New Jersey, USA, July 1994.

[232] R. John, P. Innocent, and M. Barnes. Neuro-Fuzzy Clustering of Radiographic Tibia Image Data using Type-2 Fuzzy Sets. *Information Sciences*, 125(1):65–82, 2000.

[233] I. T. Jolliffe. *Principal Component Analysis*. Springer Verlag, 1986.

[234] G. Jothi and H. H. Inbarani. Hybrid Tolerance Rough SetFirefly based Supervised Feature Selection for MRI Brain Tumor Image Classification. *Applied Soft Computing*, 46:639 – 651, 2016.

[235] D. Judd, P. Mckinley, and A. K. Jain. Large-Scale Parallel Data Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):871–876, August 1998.

[236] A. Kamburov, R. Cavill, T. M. D. Ebbels, R. Herwig, and H. C. Keun. Integrated Pathway-level Analysis of Transcriptomics and Metabolomics Data with IMPaLA. *Bioinformatics*, 27(20):2917–2918, 2011.

[237] L. Kanal. Patterns in Pattern Recognition. *IEEE Transactions on Information Theory*, 20:697–722, 1974.

[238] A. Kandal. *Fuzzy Techniques in Pattern Recognition*. New York: Wiley Interscience, 1982.

[239] N. B. Karayiannis. MECA: Maximum Entropy Clustering Algorithm. In *Proceedings of the 3rd International Fuzzy Systems Conference (FUZZ-IEEE '94)*, pages 630–635, Orlando, USA, June 1994.

[240] N. N. Karnik and J. M. Mendel. Applications of Type-2 Fuzzy Logic Systems to Forecasting of Time-Series. *Information Sciences*, 120(1):89–111, 1999.

[241] N. N. Karnik and J. M. Mendel. Centroid of a Type-2 Fuzzy Set. *Information Sciences*, 132(1):195–220, 2001.

[242] G. Karypis, E. H. Han, and V. Kumar. Chameleon: A Hierarchical Clustering Algorithm Using Dynamic Modeling. *Computer*, 32(8):68–75, 1999.

[243] G. Karystianis, K. Thayer, M. Wolfe, and G. Tsafnat. Evaluation of A Rule-based Method for Epidemiological Document Classification Towards The Automation of Systematic Reviews. *Journal of Biomedical Informatics*, 70:27 – 34, 2017.

[244] M. L. Kastens. What Good is Market Research? *IEEE Engineering Management Review*, 11(2):24–26, June 1983.

[245] I. Katakis, N. Tsapatsoulis, V. Triga, C. Tziouvas, and F. Mendez. Clustering Online Poll Data: Towards a Voting Assistance System. In *Proceedings of the 7th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP '12)*, pages 54–59, Luxembourg City, Luxembourg, December 2012.

[246] L. Ke, Z. Feng, and Z. Ren. An Efficient Ant Colony Optimization Approach to Attribute Reduction in Rough Set Theory. *Pattern Recognition Letters*, 29(9):1351–1357, 2008.

[247] E. E. Kerre and M. Nachtegael. *Fuzzy Techniques in Image Processing.* Physica-Verlag, Heidelberg, 2010.

[248] A. Kershenbaum and R. V. Slyke. Computing Minimum Spanning Trees Efficiently. In *Proceedings of the ACM Annual Conference - Volume 1 (ACM '72)*, pages 518–527, Boston, USA, 1972.

[249] A. S. Kesar. Underground Anomaly Detection by Electromagnetic Shock Waves. *IEEE Transactions on Antennas and Propagation*, 59(1):149–153, January 2011.

[250] M. M. Khan, A. Mendes, P. Zhang, and S. K. Chalup. Evolving Multi-Dimensional Wavelet Neural Networks for Classification using Cartesian Genetic Programming. *Neurocomputing*, 247:39 – 58, 2017.

[251] S. S. Kim and A. L. N. Reddy. Image-Based Anomaly Detection Technique: Algorithm, Implementation and Effectiveness. *IEEE Journal on Selected Areas in Communications*, 24(10):1942–1954, October 2006.

[252] K. Kira and L. A. Rendell. The Feature Selection Problem: Traditional Methods and A New Algorithm. In *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI '92)*, pages 129–134, San Jose, USA, July 1992.

[253] G. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall, New Delhi, India, 2005.

[254] R. Kohavi and G. H. John. Wrappers for Feature Subset Selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.

[255] T. Kohonen. *Self-Organization and Associative Memory*. Springer, Berlin, 1984.

[256] T. Kohonen. *Self-Organizing Maps*. Springer, Berlin, 2001.

[257] W. L. G. Koontz and K. Fukunaga. A Nonlinear Feature Extraction Algorithm Using Distance Transformation. *IEEE Transactions on Computers*, C-21(1):56–63, 1972.

[258] S. Kosov, F. Rottensteiner, and C. Heipke. 3D Classification of Crossroads from Multiple Aerial Images using Conditional Random Fields. In *Proceedings of the 7th IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS '12)*, pages 1–4, Tsukuba Science City, Japan, November 2012.

[259] A. Kourid and M. Batouche. A Novel Approach for Feature Selection based on MapReduce for Biomarker Discovery. In *Proceedings of the International Conference on Computer Vision and Image Analysis Applications (CVIP '15)*, pages 1–11, El Mouradi Club Sousse, Tunisia, January 2015.

[260] R. Krishnapuram, A. Joshi, O. Nasraoui, and L. Yi. Low Complexity Fuzzy Relational Clustering Algorithms for Web Mining. *IEEE Transactions on Fuzzy Systems*, 9:595–607, 2001.

[261] R. Krishnapuram and J. M. Keller. A Possibilistic Approach to Clustering. *IEEE Transactions on Fuzzy Systems*, 1(2):98–110, 1993.

[262] R. Krishnapuram and J. M. Keller. The Possibilistic C-Means Algorithm: Insights and Recommendations. *IEEE Transactions on Fuzzy Systems*, 4(3):385–393, 1996.

[263] K. L. Kroeker. Software [R]evolution: A Roundtable. *Computer*, 32(5):48–49, May 1999.

[264] F. J. Kumar, L.-C. Lim, S. P. Lim, and K. H. Lee. Nondestructive Evaluation of Large-Area PZN-8%PT Single Crystal Wafers for Medical Ultrasound Imaging Probe Applications. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 50(3):203–209, March 2003.

[265] S. Kumar, N. Jain, and S. L. Fernandes. Rough Set Based Effective Technique of Image Watermarking. *Journal of Computational Science*, 19:121 – 137, 2017.

[266] S. S. Kumar and H. H. Inbarani. Optimistic Multi-Granulation Rough Set Based Classification for Medical Diagnosis. *Procedia Computer Science*, 47:374 – 382, 2015.

[267] S. U. Kumar and H. H. Inbarani. A Novel Neighborhood Rough Set Based Classification Approach for Medical Diagnosis. *Procedia Computer Science*, 47:351 – 359, 2015.

[268] L. I. Kuncheva. *Fuzzy Classifier Design*. Physica-Verlag, Heidelberg, 2010.

[269] N. Kwak and C. H. Choi. Input Feature Selection by Mutual Information Based on Parzen Window. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(12):1667–1671, 2002.

[270] A. Lahsasna and W. C. Seng. An Improved Genetic-Fuzzy System for Classification and Data Analysis. *Expert Systems With Applications*, 83:49 – 62, 2017.

[271] M. H. C. Law, M. A. T. Figueiredo, and A. K. Jain. Simultaneous Feature Selection and Clustering using Mixture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1154–1166, 2004.

[272] D. H. Lee and M. H. Kim. Database Summarization Using Fuzzy ISA Hierarchies. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 27:68–78, 1997.

[273] T. W. Lee. *Independent Component Analysis*. Dordrech: Khrwer Academic Publishers, 1993.

[274] Y. Leung, M. M. Fischer, W.-Z. Wu, and J.-S. Mi. A Rough Set Approach for the Discovery of Classification Rules in Interval-Valued Information Systems. *International Journal of Approximate Reasoning*, 47(2):233 – 246, 2008.

[275] G.-Z. Li, S. Sun, M. You, Y.-L. Wang, and G.-P. Liu. Inquiry Diagnosis of Coronary Heart Disease in Chinese Medicine based on Symptom-Syndrome Interactions. *Chinese Medicine*, 7(1):9, 2012.

[276] H.-M. Li, C.-X. Sun, and K.-J. Wang. Clustering Web Search Results using Conceptual Grouping. In *Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC '09)*, pages 1499–1503, Baoding, China, July 2009.

[277] J. Li, Q. Du, W. Li, and Y. Li. Representation-based Hyperspectral Image Classification with Imbalanced Data. In *Proceedings of the IEEE International Geoscience*

and Remote Sensing Symposium (IGARSS '16), pages 3318–3321, Beijing, China, July 2016.

[278] L. Li, H. Jiang, W. K. Ching, and V. S. Vassiliadis. Metabolite Biomarker Discovery for Metabolic Diseases by Flux Analysis. In *Proceedings of the IEEE 6th International Conference on Systems Biology (ISB '12)*, pages 1–5, Xian, China, August 2012.

[279] X. Li, Y. Su, B. Sun, W. Ji, Z. Peng, Y. Xu, M. Wu, and C. Su. An Artificially Designed Interfering lncRNA Expressed by Oncolytic Adenovirus Competitively Consumes OncomiRs to Exert Antitumor Efficacy in Hepatocellular Carcinoma. *Molecular Cancer Therapeutics*, 15:1436–1451, 2016.

[280] Z. Li, G. Xu, M. Li, W.-Y. Ma, and H.-J. Zhang. Grouping WWW Image Search Results by Novel Inhomogeneous Clustering Method. In *Proceedings of the 11th International Multimedia Modelling Conference (MMM '05)*, pages 255–261, Melbourne, Australia, January 2005.

[281] X. Lian, Y. Wu, W. Zhao, F. Wang, Q. Zhang, and M. Li. Unsupervised SAR Image Segmentation Based on Conditional Triplet Markov Fields. *IEEE Geoscience and Remote Sensing Letters*, 11(7):1185–1189, July 2014.

[282] K. C. Liang, X. Wang, and D. Anastassiou. Bayesian Basecalling for DNA Sequence Analysis Using Hidden Markov Models. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4(3):430–440, July 2007.

[283] Q. Liang and J. M. Mendel. Interval Type-2 Fuzzy Logic Systems: Theory and Design. *IEEE Transactions on Fuzzy Systems*, 8(5):535–550, 2000.

[284] K. J. Lieberherr and C. Xiao. Object-Oriented Software Evolution. *IEEE Transactions on Software Engineering*, 19(4):313–343, April 1993.

[285] Y.-C. Lin, Y.-P. Tsai, Y.-P. Hung, and Z.-C. Shih. Comparison between Immersion-Based and Toboggan-Based Watershed Image Segmentation. *IEEE Transactions on Image Processing*, 15(3):632–640, March 2006.

[286] P. Lingras. Rough Neural Networks[C]. In *Proceedings of the 6th International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU '06)*, pages 1445–1450, Dortmund, Germany, June 1996.

[287] P. Lingras. Unsupervised Rough Set Classification Using GAs. *Journal Of Intelligent Information Systems*, 16(3):215–228, 2001.

[288] P. Lingras and C. Davies. Rough Genetic Algorithms. In *Proceedings of the 7th International Conference on New Directions in Rough Sets, Data Mining, and Granular-Soft Computing (RSFDGrC '99)*, pages 38–46, Yamaguchi, Japan, November 1999.

[289] P. Lingras and C. West. Interval Set Clustering of Web Users with Rough K-Means. *Journal of Intelligent Information Systems*, 23(1):5–16, 2004.

[290] C. Liu, D. Jiang, and W. Yang. Global Geometric Similarity Scheme for Feature Selection in Fault Diagnosis. *Expert Systems with Applications*, 41(8):3585–3595, 2014.

[291] D. Liu, S. Wang, D. Huang, G. Deng, F. Zeng, and H. Chen. Medical Image Classification using Spatial Adjacent Histogram based on Adaptive Local Binary Patterns. *Computers in Biology and Medicine*, 72:185 – 200, 2016.

[292] H. Liu and R. Setiono. A Probabilistic Approach to Feature Selection A Filter Solution. In *Proceedings of the 13th International Conference of Machine Learning (ICML '96)*, pages 319–337, Bari, Italy, 1996.

[293] H. Liu and R. Setiono. Feature Selection via Discretization of Numeric Attributes. *IEEE Transactions on Knowledge and Data Engineering*, 9(4):642–645, 1997.

[294] H. Liu, X. Yang, and R. Lu. Development of Classification Model and QSAR Model for Predicting Binding Affinity of Endocrine Disrupting Chemicals To Human Sex Hormone-Binding Globulin. *Chemosphere*, 156:1 – 7, 2016.

[295] S. Liu, T. Kawamoto, O. Morita, K. Yoshinari, and H. Honda. Discriminating Between Adaptive and Carcinogenic Liver Hypertrophy in Rat Studies using Logistic Ridge Regression Analysis of Toxicogenomic Data: The Mode of Action and Predictive Models. *Toxicology and Applied Pharmacology*, 318:79 – 87, 2017.

[296] T. Liu, S. Liu, and Z. Chen. An Evaluation on Feature Selection for Text Clustering. In *Proceedings of 20th International Conference on Machine Learning (ICML '03)*, pages 488–495, Washington DC, USA, 2003.

[297] S. P. Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28:128–137, 1982.

[298] J. Lopez-Vargas, N. Piedra, J. Chicaiza, and E. Tovar. OER Recommendation for Entrepreneurship Using a Framework Based on Social Network Analysis. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, 10(4):262–268, November 2015.

[299] D. Lowe and A. R. Webb. Optimized Feature Extraction and the Bayes Decision in Feed-Forward Classifier Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):264–355, 1991.

[300] J. A. Lozano, Q. Zhang, and P. Larranaga. Guest Editorial: Special Issue on Evolutionary Algorithms Based on Probabilistic Models. *IEEE Transactions on Evolutionary Computation*, 13(6):1197–1198, December 2009.

[301] L. Lu, J. Yan, and C. W. D. Silva. Feature Selection for ECG Signal Processing using Improved Genetic Algorithm and Empirical Mode Decomposition. *Measurement*, 94:372 – 381, 2016.

[302] X. Luo, W. Ohyama, T. Wakabayashi, and F. Kimura. A Study on Automatic Chinese Text Classification. In *Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR '11)*, pages 920–924, Beijing, China, September 2011.

[303] B. Ma and Y. Xia. A Tribe Competition-based Genetic Algorithm for Feature Selection in Pattern Classification. *Applied Soft Computing*, 58:328 – 338, 2017.

[304] H. Ma, X. Fan, and J. Chen. An Incremental Chinese Text Classification Algorithm Based on Quick Clustering. In *Proceedings of the International Symposiums on Information Processing (IPSN '08)*, pages 308–312, Missouri, USA, May 2008.

[305] J. MacQueen. Some Methods for Classification and Analysis of Multivariate Observation. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, 1967.

[306] A. Maeda, H. Ashida, Y. Taniguchi, and Y. Takahashi. Data Mining System Using Fuzzy Rule Induction. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '95)*, pages 45–46, Yokohama, Japan, May 1995.

[307] M. Maggini, L. Rigutini, and M. Turchi. Pseudo-Supervised Clustering for Text Documents. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI '04)*, pages 363–369, Beijing, China, September 2004.

[308] A. Mahapatro and P. M. Khilar. Fault Diagnosis in Wireless Sensor Networks: A Survey. *IEEE Communications Surveys Tutorials*, 15(4):2000–2026, Fourth 2013.

[309] A. Mahmood, A. Mian, and R. Owens. Semi-supervised Spectral Clustering for Image Set Classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pages 121–128, Ohio, USA, June 2014.

[310] P. Maji. f-Information Measures for Efficient Selection of Discriminative Genes from Microarray Data. *IEEE Transactions on Biomedical Engineering*, 56(4):1063–1069, 2009.

[311] P. Maji. Fuzzy-Rough Supervised Attribute Clustering Algorithm and Classification of Microarray Data. *IEEE Transactions on System, Man, and Cybernetics, Part B: Cybernetics*, 41(1):222–233, 2011.

[312] P. Maji. Rough Hypercuboid Approach for Feature Selection in Approximation Spaces. *IEEE Transactions on Knowledge and Data Engineering*, 26(1):16–29, 2014.

[313] P. Maji and P. Garai. Fuzzy-Rough MRMS Method for Relevant and Significant Attribute Selection. In *Proceedings of the 14th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU '12)*, pages 310–320, Catania, Italy, July 2012.

[314] P. Maji and P. Garai. Fuzzy-Rough Simultaneous Attribute Selection and Feature Extraction Algorithm. *IEEE Transactions on Cybernetics*, 43(4):1166–1177, 2013.

[315] P. Maji and P. Garai. On Fuzzy-Rough Attribute Selection: Criteria of Max-Dependency, Max-Relevance, Min-Redundancy, and Max-Significance. *Applied Soft Computing*, 13(9):3968–3980, 2013.

[316] P. Maji and P. Garai. IT2 Fuzzy-Rough Sets and Max Relevance-Max Significance Criterion for Attribute Selection. *IEEE Transactions on Cybernetics*, 45(8):1657–1668, 2015.

[317] P. Maji and P. Garai. Simultaneous Feature Selection and Extraction Using Feature Significance. *Fundamenta Informaticae*, 136:405–431, 2015.

[318] P. Maji, M. K. Kundu, and B. Chanda. Second Order Fuzzy Measure and Weighted Co-Occurrence Matrix for Segmentation of Brain MR Images. *Fundamenta Informaticae*, 88(1-2):161–176, 2008.

[319] P. Maji and S. K. Pal. RFCM: A Hybrid Clustering Algorithm Using Rough and Fuzzy Sets. *Fundamenta Informaticae*, 80(4):475–496, 2007.

[320] P. Maji and S. K. Pal. Rough-Fuzzy C-Medoids Algorithm and Selection of Bio-Basis for Amino Acid Sequence Analysis. *IEEE Transactions on Knowledge and Data Engineering*, 19(6):859–872, 2007.

[321] P. Maji and S. K. Pal. Rough Set Based Generalized Fuzzy C-Means Algorithm and Quantitative Indices. *IEEE Transactions on System, Man, and Cybernetics, Part B: Cybernetics*, 37(6):1529–1540, 2007.

[322] P. Maji and S. K. Pal. Maximum Class Separability for Rough-Fuzzy C-Means Based Brain MR Image Segmentation. *LNCS Transactions on Rough Sets*, 9:114–134, 2008.

[323] P. Maji and S. K. Pal. Feature Selection Using f-Information Measures in Fuzzy Approximation Spaces. *IEEE Transactions on Knowledge and Data Engineering*, 22(6):854–867, 2010.

[324] P. Maji and S. K. Pal. Fuzzy-Rough Sets for Information Measures and Selection of Relevant Genes from Microarray Data. *IEEE Transactions on System, Man, and Cybernetics, Part B: Cybernetics*, 40(3):741–752, 2010.

[325] P. Maji and S. K. Pal. *Rough-Fuzzy Pattern Recognition: Applications in Bioinformatics and Medical Imaging*. Wiley-IEEE Computer Society Press, New Jersey, Hoboken, 2012.

[326] P. Maji and S. Paul. Rough Sets for Selection of Molecular Descriptors to Predict Biological Activity of Molecules. *IEEE Transactions on System, Man, and Cybernetics, Part C: Applications and Reviews*, 40(6):639–648, 2010.

[327] P. Maji and S. Paul. Rough Set Based Maximum Relevance-Maximum Significance Criterion and Gene Selection from Microarray Data. *International Journal of Approximate Reasoning*, 52(3):408–426, 2011.

[328] P. Maji and S. Paul. Rough Set Based Feature Selection: Criteria of Max-Dependency, Max-Relevance, and Max-Significance. In A. Skowron and Z. Suraj, editors, *Rough Sets and Intelligent Systems: Professor Zdzislaw Pawlak in Memoriam*, pages 393–418. Springer, 2012.

[329] P. Maji and S. Paul. Robust Rough-Fuzzy C-Means Algorithm: Design and Applications in Coding and Non-Coding RNA Expression Data Clustering. *Fundamenta Informaticae*, 124:153–174, 2013.

[330] P. Maji and S. Paul. Rough-Fuzzy Clustering for Grouping Functionally Similar Genes from Microarray Data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 10(2):286–299, 2013.

[331] P. Maji and S. Paul. *Scalable Pattern Recognition Algorithms: Applications in Computational Biology and Bioinformatics*. Springer International Publishing, 2014.

[332] P. Maji and S. Roy. Rough-Fuzzy Clustering and Multiresolution Image Analysis for Text-Graphics Segmentation. *Applied Soft Computing*, 30:705 – 721, 2015.

[333] P. Maji and S. Roy. Rough-Fuzzy Clustering and Unsupervised Feature Selection for Wavelet Based MR Image Segmentation. *PLoS ONE*, 10(4):1 – 30, 2015.

[334] P. Maji and S. Roy. SoBT-RFW: Rough-Fuzzy Computing and Wavelet Analysis Based Automatic Brain Tumor Detection Method from MR Images. *Fundamenta Informaticae*, 142(1-4):237–267, 2015.

[335] M. Maragkakis, P. Alexiou, G. L. Papadopoulos, M. Reczko, T. Dalamagas, G. Giannopoulos, G. Goumas, E. Koukis, K. Kourtis, V. A. Simossis, P. Sethupathy, T. Vergoulis, N. Koziris, T. Sellis, P. Tsanakas, and A. G. Hatzigeorgiou. Accurate microRNA Target Prediction Correlates with Protein Repression Levels. *BMC Bioinformatics*, 10(295), 2009.

[336] J. G. Marin-Blazquez and Q. Shen. From Approximative to Descriptive Fuzzy Classifiers. *IEEE Transactions on Fuzzy Systems*, 10(4):484–497, 2002.

[337] A. M. C. Martinez, S. H. Mallidi, and B. T. Meyer. On The Relevance of Auditory-based Gabor Features for Deep Learning in Robust Speech Recognition. *Computer Speech and Language*, 45:21 – 38, 2017.

[338] L. Martnez-Sanahuja and D. Snchez. Evaluating The Suitability of Web Search Engines As Proxies for Knowledge Discovery From The Web. *Procedia Computer Science*, 96:169 – 178, 2016.

[339] B. W. Matthews. Comparison of the Predicted and Observed Secondary Structure of T4 Phage Lysozyme. *Biochimica et Biophysica Acta*, 405:442–451, 1975.

[340] L. J. Mazlack, A. He, and Y. Zhu. A Rough Set Approach in Choosing Partitioning Attributes. In *Proceedings of the ISCA 13th International Conference (CAINE '00*, pages 1–6, 2000.

[341] J. McQueen. Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the 5th Berkeley Symposium on Mathematics, Statistics and Probability*, pages 281–297, 1967.

[342] J. M. Mendel. *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Prentice-Hall, Upper Saddle River, NJ, 2001.

[343] J. M. Mendel. Computing Derivatives in Interval Type-2 Fuzzy Logic Systems. *IEEE Transactions on Fuzzy Systems*, 12(1):84–98, 2004.

[344] J. M. Mendel and R. I. John. Type-2 Fuzzy Sets Made Simple. *IEEE Transactions on Fuzzy Systems*, 10(2):117–127, 2002.

[345] J. M. Mendel, R. I. John, and F. Liu. Interval Type-2 Fuzzy Logic Systems Made Simple. *IEEE Transactions on Fuzzy Systems*, 14(6):808–821, 2006.

[346] J. M. Mendel and X. Liu. Simplified Interval Type-2 Fuzzy Logic Systems. *IEEE Transactions on Fuzzy Systems*, 21(6):1056–1069, 2013.

[347] J. M. Mendel, M. R. Rajati, and P. Sussner. On Clarifying Some Definitions and Notations used for Type-2 Fuzzy Sets as well as Some Recommended Changes. *Information Sciences*, 340(C):337–345, 2016.

[348] H. Midelfart, J. Komorowski, K. Nørsett, F. Yadetie, A. K. Sandvik, and A. Lægreid. Learning Rough Set Classifiers from Gene Expressions and Clinical Data. *Fundamenta Informaticae*, 53(2):155–183, May 2002.

[349] H. Midelfart, A. Lægreid, and J. Komorowski. *Classification of Gene Expression Data in an Ontology*, pages 186–194. 2001.

[350] J. H. Min, Y. K. Kim, S.-Y. Choi, W. K. Jeong, W. J. Lee, S. Y. Ha, S. Ahn, and H. S. Ahn. Differentiation Between Cholangiocarcinoma and Hepatocellular Carcinoma with Target Sign on Diffusion-Weighted Imaging and Hepatobiliary Phase Gadoxetic Acid-Enhanced MR Imaging: Classification Tree Analysis Applying Capsule and Septum. *European Journal of Radiology*, 92:1 – 10, 2017.

[351] H. Ming. A Rough Set Based Hybrid Method to Feature Selection. In *Proceedings of the International Symposium on Knowledge Acquisition and Modeling (KAM '08)*, pages 585–588, Wuhan, China, December 2008.

[352] M. Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA, USA, 1998.

[353] T. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[354] P. Mitra, C. A. Murthy, and S. K. Pal. Unsupervised Feature Selection using Feature Similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):301–312, 2002.

[355] S. Mitra. An Evolutionary Rough Partitive Clustering. *Pattern Recognition Letters*, 25(12):1439 – 1449, 2004.

[356] S. Mitra, H. Banka, and W. Pedrycz. Rough-Fuzzy Collaborative Clustering. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 36:795–805, 2006.

[357] S. Mitra and Y. Hayashi. Bioinformatics with Soft Computing. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 36(5):616–635, September 2006.

[358] S. Miyamoto, H. Ichihashi, and K. Honda. *Algorithms for Fuzzy Clustering: Methods in C-Means Clustering with Applications*. Springer Publishing Company, Incorporated, 2008.

[359] B. M. E. Moret. Decision Trees and Diagrams. *Computing Surveys*, 14(4):593–623, December 1982.

[360] H. Morimitsu, I. Bloch, and R. M. Cesar-Jr. Exploring Structure for Long-Term Tracking of Multiple Objects in Sports Videos. *Computer Vision and Image Understanding*, 159:89 – 104, 2017.

[361] A. Muniategui, B. Hriz, L. Eciolaza, M. Ayuso, A. Iturrioz, I. Quintana, and P. lvarez. Spot Welding Monitoring System based on Fuzzy Classification and Deep Learning. In *Proceedings of the 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '17)*, pages 1–6, July 2017.

[362] N. Murru and R. Rossini. A Bayesian Approach for Initialization of Weights in Backpropagation Neural Net with Application To Character Recognition. *Neurocomputing*, 193:92 – 105, 2016.

[363] T. Nagel, J. Klerkx, A. V. Moere, and E. Duval. *Unfolding - A Library for Interactive Maps*. Springer, Berlin, Heidelberg, 2013.

[364] S. Nagy, F. Lilik, and L. T. Koczy. Entropy based Fuzzy Classification and Detection Aid for Colorectal Polyps. In *Proceedings of the 2017 IEEE African Conference (AFRICON '17)*, pages 78–82, September 2017.

[365] A. Namburu, S. K. Samay, and S. R. Edara. Soft Fuzzy Rough Set-Based MR Brain Image Segmentation. *Applied Soft Computing*, 54:456 – 466, 2017.

[366] O. Nasraoui, R. Krishnapuram, and A. Joshi. Relational Clustering Based on a New Robust Estimator with Application to Web Mining. In *Proceedings of the 18th International Conference of the North American Fuzzy Information Processing Society (NAFIPS '99)*, pages 705–709, New York, 1999.

[367] M. Ng and L. Chan. Informative Gene Discovery for Cancer Classification from Microarray Expression Data. In *Proceedings of the IEEE Workshop on Machine Learning for Signal Processing (MLSP '05)*, pages 393–398, Connecticut, USA, September 2005.

[368] R. T. Ng and J. Han. Efficient and Effective Clustering Methods for Spatial Data Mining. In *Proceedings of the 20th International Conference on Very Large Data Bases*, pages 144–155, Santiago, Chile, 1994.

[369] N. Nguyen, Q. M. J. Wu, and S. Ahuja. An Extension of the Standard Mixture Model for Image Segmentation. *IEEE Transactions on Neural Networks*, 21(8):1326–1338, August 2010.

[370] S. Nguyen, T. Nguyen, A. Skowron, and P. Synak. Knowledge Discovery by Rough Set Methods. In *Proceedings of the International Conference on Information Systems Analysis and Synthesis (ISAS '96)*, pages 26–33, Orlando, USA, July 1996.

[371] T. Nguyen, S. Nahavandi, D. Creighton, and A. Khosravi. Mass Spectrometry Cancer Data Classification using Wavelets and Genetic Algorithm. *FEBS Letters*, 589(24, Part B):3879 – 3886, 2015.

[372] D. Nicorici, J. Astola, and O. Yli-Harja. Estimating the Number of Clusters in Microarray Data Sets based on an Information Theoretic Criterion. In *Proceedings of the IEEE/SP 13th Workshop on Statistical Signal Processing (SSP '05)*, pages 1002–1007, Bordeaux, France, July 2005.

[373] F. M. Noori, N. Naseer, N. K. Qureshi, H. Nazeer, and R. A. Khan. Optimal Feature Selection from Fnirs Signals using Genetic Algorithms for BCI. *Neuroscience Letters*, 647:61 – 66, 2017.

[374] R. Nowicki. On Combining Neuro-Fuzzy Architectures with the Rough Set Theory to Solve Classification Problems with Incomplete Data. *IEEE Transactions on Knowledge and Data Engineering*, 20(9):1239–1253, 2008.

[375] S. Nuske, J. Roberts, and G. Wyeth. Extending the Dynamic Range of Robotic Vision. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '06)*, pages 162–167, May 2006.

[376] J. T. Oliva, H. D. Lee, N. Spolar, C. S. R. Coy, and F. C. Wu. Prototype System for Feature Extraction, Classification and Study of Medical Images. *Expert Systems with Applications*, 63:267 – 283, 2016.

[377] T. F. Oliver, B. Schmidt, Y. Jakop, and D. L. Maskell. High Speed Biological Sequence Analysis With Hidden Markov Models on Reconfigurable Platforms. *IEEE Transactions on Information Technology in Biomedicine*, 13(5):740–746, September 2009.

[378] J. I. Olszewska. Active Contour based Optical Character Recognition for Automated Scene Understanding. *Neurocomputing*, 161:65 – 71, 2015.

[379] C. OReilly, A. Gluhak, and M. A. Imran. Adaptive Anomaly Detection with Kernel Eigenspace Splitting and Merging. *IEEE Transactions on Knowledge and Data Engineering*, 27(1):3–16, January 2015.

[380] F. Pacheco, M. Cerrada, R. V. Snchez, D. Cabrera, C. Li, and J. V. D. Oliveira. Clustering Algorithm Using Rough Set Theory for Unsupervised Feature Selection. In *Proceedings of the International Joint Conference On Neural Networks (IJCNN '16)*, pages 3493–3499, July 2016.

[381] F. Pacheco, M. Cerrada, R.-V. Snchez, D. Cabrera, C. Li, and J. V. D. Oliveira. Attribute Clustering using Rough Set Theory for Feature Selection in Fault Severity Classification of Rotating Machinery. *Expert Systems With Applications*, 71:69 – 86, 2017.

[382] A. Pal and S. K. Pal. Pattern Recognition: Evolution of Methodologies and Data Mining. In S. K. Pal and A. Pal, editors, *Pattern Recognition: from Classical to Modern Approaches*, pages 1–23. World Scientific, Singapore, 2001.

[383] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek. A Possibilistic Fuzzy C-Means Clustering Algorithm. *IEEE Transactions on Fuzzy Systems*, 13(4):517–530, 2005.

[384] S. K. Pal. Soft Computing Pattern Recognition: Principles, Integrations and Data Mining. In T. Terano, T. Nishida, A. Namatame, S. Tsumoto, Y. Ohswa, and T. Washio, editors, *Advances in Artificial Intelligence*, volume 2253 of *Lecture Notes in Artificial Intelligence*, pages 261–268. Springer-Verlag, Berlin, 2002.

[385] S. K. Pal, R. K. De, and J. Basak. Unsupervised Feature Evaluation: A Neuro-Fuzzy Approach. *IEEE Transactions on Neural Network*, 11(2):366–376, 2000.

[386] S. K. Pal, T. S. Dillon, and D. S. Yeung. *Soft Computing in Case Based Reasoning*. Springer-Verlag, London, 2000.

[387] S. K. Pal and A. Ghosh. Index of Area Coverage of Fuzzy Image Subsets and Object Extraction. *Pattern Recognition Letters*, 11(12):831–841, 1990.

[388] S. K. Pal and A. Ghosh. Image Segmentation Using Fuzzy Correlation. *Information Sciences*, 62(3):223–250, 1992.

[389] S. K. Pal, A. Ghosh, and B. U. Shankar. Segmentation of Remotely Sensed Images with Fuzzy Thresholding and Quantitative Evaluation. *International Journal of Remote Sensing*, 21(11):2269–2300, 2000.

[390] S. K. Pal, R. A. King, and A. A. Hashim. Automatic Gray Level Thresholding Through Index of Fuzziness and Entropy. *Pattern Recognition Letters*, (1):141–146, 1983.

[391] S. K. Pal and D. D. Majumder. *Fuzzy Mathematical Approach to Pattern Recognition*. Wiley, Halsted Press, New York, 1986.

[392] S. K. Pal, S. Meher, and S. Dutta. Class-Dependent Rough-Fuzzy Granular Space, Dispersion Index and Classification. *Pattern Recognition*, 45(7):2690–2707, 2012.

[393] S. K. Pal and P. Mitra. *Pattern Recognition Algorithms for Data Mining*. CRC Press, Boca Raton, Florida, 2004.

[394] S. K. Pal and S. Mitra. *Neuro-Fuzzy Pattern Recognition: Methods in Soft Computing*. New York: John Wiley & Sons, 1999.

[395] S. K. Pal and P. K. Pramanik. Fuzzy Measures in Determining Seed Points in Clustering. *Pattern Recognition Letters*, 4(3):159–164, 1986.

[396] B. Papari, C. S. Edrington, and F. Kavousi-Fard. An Effective Fuzzy Feature Selection and Prediction Method for Modeling Tidal Current: A Case of Persian Gulf. *IEEE Transactions on Geoscience and Remote Sensing*, 55(9):4956–4961, September 2017.

[397] I.-K. Park and G.-S. Choi. A Variable-Precision Informationentropy Rough Set Approach for Job Searching. *Information Systems*, 48:279–288, 2015.

[398] I.-K. Park and G.-S. Choi. Rough Set Approach for Clustering Categorical Data using Information-Theoretic Dependency Measure. *Information Systems*, 48:289 – 295, 2015.

[399] H. Parkinson, M. Kapushesky, N. Kolesnikov, G. Rustici, M. Shojatalab, N. Abeygunawardena, H. Berube, M. Dylag, I. Emam, A. Farne, E. Holloway, M. Lukk, J. Malone, R. Mani, E. Pilicheva, T. F. Rayner, F. Rezwan, A. Sharma, E. Williams, X. Z. Bradley, T. Adamusiak, M. Brandizi, T. Burdett, R. Coulson, M. Krestyaninova, P. Kurnosov, E. Maguire, S. G. Neogi, P. R. Serra, S. A. Sansone, N. Sklyar, M. Zhao, U. Sarkans, and A. Brazma. ArrayExpress Update - From an Archive of Functional Genomics Experiments to the Atlas of Gene Expression. *Nucleic Acids Research*, 27:D865–D872, 2009.

[400] N. Parthalain, Q. Shen, and R. Jensen. A Distance Measure Approach to Exploring the Rough Set Boundary Region for Attribute Reduction. *IEEE Transactions on Knowledge and Data Engineering*, 22(3):305–317, 2010.

[401] S. Pasala, B. N. Kumar, and S. C. Satapathy. *A Study of Roulette Wheel and Elite Selection on GA to Solve Job Shop Scheduling*, pages 477–485. January 2013.

[402] K. R. Pattipati and M. G. Alexandridis. Application of Heuristic Search and Information Theory to Sequential Fault Diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(4):872–887, July 1990.

[403] D. Paul, R. Su, M. Romain, V. Sbastien, V. Pierre, and G. Isabelle. Feature Selection for Outcome Prediction in Oesophageal Cancer using Genetic Algorithm and Random forest Classifier. *Computerized Medical Imaging and Graphics*, pages 1 – 12, 2016.

[404] P. P. Paul, M. L. Gavrilova, and R. Alhajj. Decision Fusion for Multimodal Biometrics Using Social Network Analysis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(11):1522–1533, November 2014.

[405] S. Paul, P. Lakatos, A. Hartmann, R. Schneider-Stock, and J. Vera. Identification of miRNA-mRNA Modules in Colorectal Cancer Using Rough Hypercuboid Based Supervised Clustering. *Scientific Reports*, 7(42809):1 – 18, 2017.

[406] S. Paul and P. Maji. Robust RFCM Algorithm for Identification of Co-Expressed miRNAs. In *Proceedings of IEEE International Conference on Bioinformatics and Biomedicine*, pages 520–523, Philadelphia, USA, 2012.

[407] S. Paul and P. Maji. Rough Sets for Insilico Identification of Differentially Expressed miRNAs. *International Journal of Nanomedicine*, 8:63–74, 2013.

[408] S. Paul and P. Maji. City Block Distance and Rough-Fuzzy Clustering for Identification of Co-Expressed microRNAs. *Molecular BioSystems*, 10:1509–1523, 2014.

[409] S. Paul and P. Maji. City Block Distance and Rough-Fuzzy Clustering for Identification of Co-Expressed microRNAs. *Molecular BioSystems*, 10:1509–1523, 2014.

[410] S. Paul and P. Maji. Gene Ontology based Quantitative Index to Select Functionally Diverse Genes. *International Journal of Machine Learning and Cybernetics*, 5(2):245–262, 2014.

[411] S. Paul and P. Maji. Gene Expression and Protein–Protein Interaction Data for Identification of Colon Cancer Related Genes using f-Information Measures. *Natural Computing*, 15(3):449–463, 2016.

[412] T. Pavlidis. *Structural Pattern Recognition.* Springer-Verlag, New York, 1977.

[413] Z. Pawlak. Rough Sets. *International Journal of Computer and Information Science*, 11:341–356, 1982.

[414] Z. Pawlak. *Rough Sets: Theoretical Aspects of Resoning About Data.* Kluwer, Dordrecht, The Netherlands, 1991.

[415] W. Pedrycz. Shadowed Sets: Bridging Fuzzy and Rough Sets. In S. K Pal and A. Skowron, editors, *Rough Fuzzy Hybridization. A New Trend in Decision-Making*, pages 179–199, Singapore, 1998. Springer-Verlag.

[416] H. Peng, F. Long, and C. Ding. Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[417] E. Perales, J. I. Garca, E. Pires, L. Aldea, L. Lomba, and B. Giner. Ecotoxicity and QSAR Studies of Glycerol Ethers in Daphnia Magna. *Chemosphere*, 183:277 – 285, 2017.

[418] L. I. Perlovsky. Conundrum of Combinatorial Complexity. *IEEE Transaction Pattern Analysis and Machine Intelligence*, 20(6):666–670, 1998.

[419] 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*, 54(2):307–322, 2013.

[420] T. V. Phan and M. Nakagawa. Combination of Global and Local Contexts for Text/Non-Text Classification in Heterogeneous Online Handwritten Documents. *Pattern Recognition*, 51:112 – 124, 2016.

[421] M. Piccioni, M. Oriol, and B. Meyer. Class Schema Evolution for Persistent Object-Oriented Software: Model, Empirical Study, and Automated Support. *IEEE Transactions on Software Engineering*, 39(2):184–196, February 2013.

[422] H. Pirim. *A Minimum Spanning Tree Based Clustering Algorithm for High Throughput Biological Data*. PhD thesis, Mississippi State, MS, USA, 2011.

[423] F. Pourpanah, C. P. Lim, and J. M. Saleh. A Hybrid Model of Fuzzy ARTMAP and Genetic Algorithm for Data Classification and Rule Extraction. *Expert Systems with Applications*, 49:74 – 85, 2016.

[424] D. M. W. Powers. Evaluation: From Precision, Recall and f-measure to ROC., Informedness, Markedness and Correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[425] P. Prashant, A. Joshi, and V. Gandhi. Brain Computer Interface: A Review. In *Proceedings of the 5th Nirma University International Conference on Engineering (NUiCONE '15)*, pages 1–6, Ahmedabad, India, November 2015.

[426] A. Prez-Garrido, F. Girn-Rodrguez, A. Bueno-Crespo, J. Soto, H. Prez-Snchez, and A. M. Helguera. Fuzzy Clustering As Rational Partition Method for QSAR. *Chemometrics and Intelligent Laboratory Systems*, 166:1 – 6, 2017.

[427] Y. Qian, J. Liang, W. Pedrycz, and C. Dang. Positive Approximation: An Accelerator for Attribute Reduction in Rough Set Theory. *Artificial Intelligence*, 174(9–10):597–618, 2010.

[428] S. Qiu, Z. Li, W. He, L. Zhang, C. Yang, and C. Y. Su. Brain-Machine Interface and Visual Compressive Sensing-Based Teleoperation Control of an Exoskeleton Robot. *IEEE Transactions on Fuzzy Systems*, 25(1):58–69, February 2017.

[429] I. A. Questier, B. W. Rollier, and D. L. Massart. Application of Rough Set Theory To Feature Selection For Unsupervised Clustering. *Chemometrics And Intelligent Laboratory Systems*, 63:155–167, 2002.

[430] J. R. Quinlan. Induction of Decision Trees. *Machine Learning*, 1:81–106, 1986.

[431] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, CA, 1993.

[432] V. Raijlich, N. Wilde, M. Buckellew, and H. Page. Software Cultures and Evolution. *Computer*, 34(9):24–28, September 2001.

[433] W. M. Rand. Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[434] P. M. Raposeiro, A. C. Costa, and S. J. Hughes. Environmental Factors – Spatial and Temporal Variation of Chironomid Communities in Oceanic Island Streams (Azores Archipelago). *International Journal of Limnology*, 47(4):325–338, 2011.

[435] W. E. Reddick, J. O. Glass, E. N. Cook, T. D. Elkin, and R. J. Deaton. Automated Segmentation and Classification of Multispectral Magnetic Resonance Images of Brain using Artificial Neural Networks. *IEEE Transactions on Medical Imaging*, 16(6):911–918, December 1997.

[436] F. C. H. Rhee. Uncertain Fuzzy Clustering: Insights and Recommendations. *IEEE Computational Intelligence Magazine*, 2(1):44–56, 2007.

[437] F. C. H. Rhee and C. Hwang. A Type-2 Fuzzy C-Means Clustering Algorithm. In *Joint 9th IFSA World Congress and 20th NAFIPS International Conference (IFSA-NAFIPS '01)*, volume 4, pages 1926–1929, Vancouver, Canada, July 2001.

[438] F. C. H. Rhee and R. Krishnapuram. Fuzzy Rule Generation Methods for High-Level Computer Vision. *Fuzzy Sets and Systems*, 60(3):245–258, 1993.

[439] B. Ripley. Statistical Aspects of Neural Networks. In *Networks on Chaos: Statistical and Probabilistic Aspects*. Chapman and Hall, 1993.

[440] P. Rocchi. An Essay on the Origin of Software Evolution. *IEEE Transactions on Human-Machine Systems*, 44(2):281–285, April 2014.

[441] F. J. Rohlf. Single Link Clustering Algorithms. In *Handbook of Statistics*, pages 267–284. North-Holland, Amsterdam, 1982.

[442] A. Rohrbeck, F. Stahl, M. Holtje, T. Hettwer, P. Lindner, S. Hagemann, A. Pich, and K. Haastert-Talini. C3-Induced Release of Neurotrophic Factors from Schwann Cells - Potential Mechanism Behind its Regeneration Promoting Activity. *Neurochemistry International*, 90:232–245, 2015.

[443] L. Rokach and O. Maimon. *Clustering Methods*, pages 321–352. Springer US, 2005.

[444] C. Romero and S. Ventura. Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(6):601–618, November 2010.

[445] M. Roubens. Pattern Classification Problems and Fuzzy Sets. *Fuzzy Sets and Systems*, 1:239–253, 1978.

[446] J. P. Rousseeuw. Silhouettes: A Graphical Aid to the Interpration and Validation of Cluster Analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[447] 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. *Advances in Fuzzy Systems*, 17(2):1–23, 2017.

[448] L. Rui, D. Zhijiang, and S. Lining. Moving Object Tracking based on Mobile Robot Vision. In *Proceedings of the International Conference on Mechatronics and Automation (IMCA '09)*, pages 3625–3630, Takamatsu, Japan, August 2009.

[449] B. Rujiang and L. Junhua. A Novel Conception Based Texts Classification Method. In *Proceedings of the International e-Conference on Advanced Science and Technology (AST '09)*, pages 30–34, Daejeon, South Korea, March 2009.

[450] E. H. Ruspini. A New Approach to Clustering. *Information and Control*, 15:22–32, 1969.

[451] M. Russo. FuGeNeSys: A Fuzzy Genetic Neural System for Fuzzy Modeling. *IEEE Transactions on Fuzzy Systems*, 6(3):373–388, 1998.

[452] A. Sadhu and B. Bhattacharyya. Discovery of Cancer Linked Biomarker Genes through Common Subcluster Mining. In *Proceedings of the International Conference*

on Bioinformatics and Systems Biology (BSB '16), pages 1–5, Allahabad, India, March 2016.

[453] I. Saha, J. P. Sarkar, and U. Maulik. Ensemble Based Rough Fuzzy Clustering for Categorical Data. *Knowledge-Based Systems*, 77:114 – 127, 2015.

[454] R. Saltos and R. Weber. A Rough-Fuzzy Approach for Support Vector Clustering. *Information Sciences*, 339:353 – 368, 2016.

[455] J. W. Sammon. A Nonlinear Mapping for Data Structure Analysis. *IEEE Transactions on Computer*, 18:401–409, 1969.

[456] A. C. Santos, A. C. B. Delbem, J. B. A. London, and N. G. Bretas. Node-Depth Encoding and Multiobjective Evolutionary Algorithm Applied to Large-Scale Distribution System Reconfiguration. *IEEE Transactions on Power Systems*, 25(3):1254–1265, August 2010.

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

[458] I. Ben Sassi, S. Ben Yahia, and S. Mellouli. Fuzzy Classification-Based Emotional Context Recognition from Online Social Networks Messages. In *Proceedings of the 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '17)*, pages 1–6, July 2017.

[459] S. M. Savaresi, D. Boley, S. Bittanti, and G. Gazzaniga. Cluster selection in divisive clustering algorithms. In *Proceedings of the Second SIAM International Conference on Data Mining (SIAM '02)*, pages 299–314, Arlington, USA, April 2002.

[460] R. J. Schalkoff. *Pattern Recognition: Statistical, Structural and Neural Approaches*. John Wiley and Sons., 1992.

[461] B. Scholkopf, A. Smola, and K. R. Muller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, 10:1299–1319, 1998.

[462] H.-P. Schwefel. *Numerical Optimization of Computer Models*. John Wiley and Sons, Inc., New York, USA, 1981.

[463] S. M. Schweizer and J. M. F. Moura. Hyperspectral Imagery: Clutter Adaptation in Anomaly Detection. *IEEE Transactions on Information Theory*, 46(5):1855–1871, August 2000.

[464] G. S. Sebestyen. *Decision Making Processes in Pattern Recognition.* New York: Macmillan, 1962.

[465] P. Sen and G. K. Mehta. Application of Proton Induced X-Ray Emission (PIXE) Technique to Gunshot Residue Analyses and Other Crime Related Problems. *IEEE Transactions on Nuclear Science*, 30(2):1302–1304, April 1983.

[466] C. H. Seng, M. G. Amin, F. Ahmad, and A. Bouzerdoum. Image Segmentations for Through-the-Wall Radar Target Detection. *IEEE Transactions on Aerospace and Electronic Systems*, 49(3):1869–1896, July 2013.

[467] P. Sethi and C. Leangsuksun. A Novel Computational Framework for Fast Distributed Computing and Knowledge Integration for Microarray Gene Expression Data Analysis. In *Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA '06)*, pages 1–5, Vienna, Austria, April 2006.

[468] J. L. Sevilla, V. Segura, A. Podhorski, E. Guruceaga, J. M. Mato, L. A. Martinez-Cruz, F. J. Corrales, and A. Rubio. Correlation Between Gene Expression and GO Semantic Similarity. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2(4):330–338, 2005.

[469] M. H. Sharif. A Numerical Approach for Tracking Unknown Number of Individual Targets in Videos. *Digital Signal Processing*, 57:106 – 127, 2016.

[470] G. Sheikholeslami, S. Chatterjee, and A. Zhang. WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases. In *Proceedings of the 24rd International Conference on Very Large Data Bases (VLDB '98)*, pages 428–439, New York, USA, August 1998.

[471] A. Sheinker, L. Frumkis, B. Ginzburg, N. Salomonski, and B. Z. Kaplan. Magnetic Anomaly Detection Using a Three-Axis Magnetometer. *IEEE Transactions on Magnetics*, 45(1):160–167, January 2009.

[472] Z. Shen, K.-L. Ma, and T. Eliassi-Rad. Visual Analysis of Large Heterogeneous Social Networks by Semantic and Structural Abstraction. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1427–1439, November 2006.

[473] T. Siegfried, S. Bleuler, M. Laumanns, E. Zitzler, and W. Kinzelbach. Multiobjective Groundwater Management Using Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation*, 13(2):229–242, April 2009.

[474] M. Sinaee and E. G. Mansoori. Fuzzy Rule Based Clustering for Gene Expression Data. In *Proceedings of the 4th International Conference on Intelligent Systems, Modelling and Simulation (ISMS '13)*, pages 7–11, Bangkok, Thailand, January 2013.

[475] A. Singh and K. K. Singh. Satellite Image Classification using Genetic Algorithm Trained Radial Basis Function Neural Network, Application To the Detection of Flooded Areas. *Journal of Visual Communication and Image Representation*, 42:173 – 182, 2017.

[476] R. Smierzchalski and Z. Michalewicz. Modeling of Ship Trajectory in Collision Situations by an Evolutionary Algorithm. *IEEE Transactions on Evolutionary Computation*, 4(3):227–241, September 2000.

[477] T. Srensen. A Method of Establishing Groups of Equal Amplitude in Plant Sociology based on Similarity of Species and its Application to Analyses of the Vegetation on Danish Commons. *Kongelige Danske Videnskabernes Selskab*, 5(4):1–34, 1948.

[478] P. Srinivasan, M. E. Ruiz, D. H. Kraft, and J. Chen. Vocabulary Mining for Information Retrieval: Rough Sets and Fuzzy Sets. *Information Processing and Management*, 37(1):15–38, 2001.

[479] I. Steinwart and A. Christmann. *Support Vector Machines*. Springer Publishing Company, Incorporated, 2008.

[480] C. Sui, R. Togneri, and M. Bennamoun. A Cascade Gray-Stereo Visual Feature Extraction Method for Visual and Audio-Visual Speech Recognition. *Speech Communication*, 90:26 – 38, 2017.

[481] A. Sun, E. P. Lim, W. K. Ng, and J. Srivastava. Blocking reduction strategies in hierarchical text classification. *IEEE Transactions on Knowledge and Data Engineering*, 16(10):1305–1308, October 2004.

[482] A. X. Sun and X. Y. Yu. Improved SOM Algorithm-EDSOM Applied in Text Clustering. In *Proceedings of the International Conference on Computer Application and System Modeling (ICCASM '10)*, pages 42–44, Taiyuan, China, October 2010.

[483] F. Sun, D. Morris, W. Lee, M. J. Taylor, T. Mills, and P. S. Babyn. Feature-Space-Based fMRI Analysis Using the Optimal Linear Transformation. *IEEE Transactions on Information Technology in Biomedicine*, 14(5):1279–1290, September 2010.

[484] J. Sun, G. L. Zhang, S. Li, A. R. Ivanov, D. Fenyo, F. Lisacek, S. K. Murthy, B. L. Karger, and V. Brusic. Pathway Analysis and Transcriptomics Improve Protein

Identification by Shotgun Proteomics from Samples Comprising Small Number of Cells - A Benchmarking Study. *BMC Genomics*, 15(9):S1, 2014.

[485] S. Sun, B. Zhang, L. Xie, and Y. Zhang. An Unsupervised Deep Domain Adaptation Approach for Robust Speech Recognition. *Neurocomputing*, 257:79 – 87, 2017.

[486] K. P. Swetha and V. S. Devi. Simultaneous Feature Selection and Clustering Using Particle Swarm Optimization. In *Neural Information Processing*, volume 7663 of *Lecture Notes in Computer Science*, pages 509–515. 2012.

[487] R. W. Swiniarski and A. Skowron. Rough Set Methods in Feature Selection and Recognition. *Pattern Recognition Letters*, 24(6):833–849, 2003.

[488] M. Y. SyarifahAdilah, R. Abdullah, and I. Venkat. ABC Algorithm as Feature Selection for Biomarker Discovery in Mass Spectrometry Analysis. In *Proceedings of the 4th Conference on Data Mining and Optimization (DMO '12)*, pages 67–72, Langkawi, Malaysia, September 2012.

[489] H. Tang, T. Li, T. Qiu, and Y. Park. Segmentation of Heart Sounds based on Dynamic Clustering. *Biomedical Signal Processing and Control*, 7(5):509 – 516, 2012.

[490] L. Tang and X. Wang. A Hybrid Multiobjective Evolutionary Algorithm for Multiobjective Optimization Problems. *IEEE Transactions on Evolutionary Computation*, 17(1):20–45, February 2013.

[491] S. Tavazoie, D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church. Systematic Determination of Genetic Network Architecture. *Nature Genetics*, 22(3):281–285, 1999.

[492] N. T. Thasleena and S. C. Varghese. Enhanced Associative Classification of XML Documents Supported By Semantic Concepts. *Procedia Computer Science*, 46:194 – 201, 2015.

[493] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Elsevier Science, 2008.

[494] H. Thiele. Fuzzy Rough Sets versus Rough Fuzzy Sets - An Interpretation and a Comparative Study using concepts of Modal Logics. *Technical report no. CI-30/98*, 1998.

[495] R. Tibshirani. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[496] O. J. Tobias and R. Seara. Image Segmentation by Histogram Thresholding Using Fuzzy Sets. *IEEE Transactions on Image Processing*, 11(12):1457–1465, 2002.

[497] P. Tomer and A. Goyal. Ant Clustering based Text Detection in Natural Scene Images. In *Proceedings of the 4th International Conference on Computing, Communications and Networking Technologies (ICCCNT '13)*, pages 1–7, Tiruchengode, India, July 2013.

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

[499] S. Trabelsi, Z. Elouedi, and P. Lingras. Classification Systems based on Rough Sets under the Belief Function Framework. *International Journal of Approximate Reasoning*, 52(9):1409 – 1432, 2011.

[500] E. C. C. Tsang, D. Chen, D. S. Yeung, X.-Z. Wang, and J. Lee. Attributes Reduction Using Fuzzy Rough Sets. *IEEE Transactions on Fuzzy Systems*, 16(5):1130–1141, 2008.

[501] E. C. C. Tsang, D. S. Yeung, and X. Z. Wang. OFFSS: Optimal Fuzzy-Valued Feature Subset Selection. *IEEE Transactions on Fuzzy Systems*, 11(2):202–213, April 2003.

[502] T.-L. Tseng, C.-C. Huang, K. Fraser, and H.-W. Ting. Rough Set Based Rule Induction in Decision Making using Credible Classification and Preference from Medical Application Perspective. *Computer Methods and Programs in Biomedicine*, 127:273 – 289, 2016.

[503] P. Tsitou, M. Heneweer, and P. J. Boogaard. Toxicogenomics in Vitro As An Alternative Tool for Safety Evaluation of Petroleum Substances and PAHS with Regard To Prenatal Developmental Toxicity. *Toxicology in Vitro*, 29(2):299 – 307, 2015.

[504] A. P. Tsou, Y. M. Sun, C. L. Liu, H. D. Huang, J. T. Horng, M. F. Tsai, and B. J. Liu. Biological Data Warehousing System for Identifying Transcriptional Regulatory Sites From Gene Expressions of Microarray Data. *IEEE Transactions on Information Technology in Biomedicine*, 10(3):550–558, July 2006.

[505] S. Ubukata, A. Notsu, and K. Honda. The Rough Set K-Means Clustering. In *Proceedings of the Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS '16) and 17th International Symposium On Advanced Intelligent Systems (ISIS '16)*, pages 189–193, August 2016.

[506] G. Uchyigit. Experimental Evaluation of Feature Selection Methods for Text Classification. In *Proceedings of the 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '12)*, pages 1294–1298, Sichuan, China, May 2012.

[507] G. Vachtsevanos, Y. Ding, J. A. Fairley, A. B. Gardner, and P. Simeonova. Microarray Gene Expression Data Analysis. In *Proceedings of the 2nd IEEE International Symposium on Biomedical Imaging: Nano to Macro (ISBI '04)*, pages 105–108, Arlington, Virginia, April 2004.

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

[509] J. J. Verschuur and A. A. Potjer. Maximization of the Economic Value of Research Through Quantifying the Relations between Research and the Market. *IEEE Transactions on Engineering Management*, EM-21(4):115–118, November 1974.

[510] I. B. Vidinli and R. Ozcan. New Query Suggestion Framework and Algorithms: A Case Study for An Educational Search Engine. *Information Processing and Management*, 52(5):733 – 752, 2016.

[511] V. E. Vinzi, W. W. Chin, J. Henseler, and H. Wang. *Handbook of Partial Least Squares*. Springer, 2010.

[512] E. H. Vogel. Market Research. *Proceedings of the Institute of Radio Engineers (IRE '50)*, 38(8):859–859, August 1950.

[513] D. Waillaeys and N. Malvache. The Use of Fuzzy Sets for the Treatment of Fuzzy Information by Computer. *Fuzzy Sets and Systems*, pages 323–328, 1981.

[514] G. S. Walia, S. Raza, A. Gupta, R. Asthana, and K. Singh. A Novel Approach of Multi-Stage Tracking for Precise Localization of Target in Video Sequences. *Expert Systems with Applications*, 78:208 – 224, 2017.

[515] D. Wang, Z. Zhang, R. Bai, and Y. Mao. A Hybrid System with Filter Approach and Multiple Population Genetic Algorithm for Feature Selection in Credit Scoring. *Journal of Computational and Applied Mathematics*, pages 1 – 14, 2017.

[516] H. Wang, Z. Wang, X. Li, B. Gong, L. Feng, and Y. Zhou. A Robust Approach Based on Weibull Distribution for Clustering Gene Expression Data. *Algorithms for Molecular Biology*, 6(1):14, 2011.

[517] J. Wang, X.-Y. Peng, and Y. Peng. Efffcient Rough-Set Based Attribute Reduction Algorithm with Nearest Neighbour Searching. *Electronics Letters*, 43(10):563 – 564, 2007.

[518] J. Wang, P. Zhao, S. C. H. Hoi, and R. Jin. Online Feature Selection and Its Applications. *IEEE Transactions on Knowledge and Data Engineering*, 26(3):698–710, 2014.

[519] L. X. Wang. A New Look at Type-2 Fuzzy Sets and Type-2 Fuzzy Logic Systems. *IEEE Transactions on Fuzzy Systems*, 25(3):693–706, June 2017.

[520] W. Wang, J. Yang, and R. Muntz. STING: A Statistical Information Grid Approach to Spatial Data Mining. In *Proceedings of the International Conference of Very Large Databases (VLDB '97)*, pages 186–195, Athens, Greece, August 1997.

[521] W.-Y. Wang and Y.-H. Li. Evolutionary Learning of BMF Fuzzy-Neural Networks Using a Reduced-Form Genetic Algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 33(6):966–976, 2003.

[522] X. Wang, X. Chen, W. Han, A. Ruan, L. Chen, R. Wang, Z. Xu, P. Xiao, X. Lu, Y. Zhao, J. Zhou, S. Chen, Q. Du, H. Yang, and X. Zhang. miR-200c Targets CDK2 and Suppresses Tumorigenesis in Renal Cell Carcinoma. *Neurochemistry International*, 13:1567–1577, 2015.

[523] X. Wang and C. Dong. Improving Generalization of Fuzzy IF-THEN Rules by Maximizing Fuzzy Entropy. *IEEE Transactions on Fuzzy Systems*, 17(3):556–567, 2009.

[524] X. Wang, L. Dong, and J. Yan. Maximum Ambiguity Based Sample Selection in Fuzzy Decision Tree Induction. *IEEE Transactions on Knowledge and Data Engineering*, 24(8):1491–1505, 2012.

[525] H.-L. Wei and S. A. Billings. Feature Subset Selection and Ranking for Data Dimensionality Reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1):162–166, 2007.

[526] J.-M. Wei, S.-Q. Wang, and X.-J. Yuan. Ensemble Rough Hypercuboid Approach for Classifying Cancers. *IEEE Transactions on Knowledge and Data Engineering*, 22(3):381–391, 2010.

[527] Q. Wei and G. Chen. Mining Generalized Association Rules with Fuzzy Taxonomic Structures. In *Proceedings of the 18th International Conference of the North American Fuzzy Information Processing Society (NAFIPS '99)*, pages 477–481, New York, 1999.

[528] P. H. Winston. *Learning Structural Descriptions from Examples*. McGraw Hill, New York, 1975.

[529] J. Wroblewski. Finding Minimal Reducts using Genetic Algorithms. *Technical Report, Institute of Mathematics, University of Warsaw*, 16(95), 1995.

[530] H. Wu and L. He. Combining Visual and Textual Features for Medical Image Modality Classification with pNorm Multiple Kernel Learning. *Neurocomputing*, 147:387 – 394, 2015.

[531] H. Wu, Y. Wu, and J. Luo. An Interval Type-2 Fuzzy Rough Set Model for Attribute Reduction. *IEEE Transactions on Fuzzy Systems*, 17(2):301–315, 2009.

[532] X. Wu, Z. Wu, and K. Li. Classification and Identification of Differential Gene Expression for Microarray Data: Improvement of the Random Forest Method. In *Proceedings of the 2nd International Conference on Bioinformatics and Biomedical Engineering (iCBBE '08)*, pages 763–766, Shanghai, China, May 2008.

[533] X. Wu, K. Yu, W. Ding, H. Wang, and X. Zhu. Online Feature Selection with Streaming Features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(5):1178–1192, 2013.

[534] Z. Wu and R. Leahy. An Optimal Graph Theoretic Approach to Data Clustering: Theory and Its Application to Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1101–1113, 1993.

[535] Z. Wu, H. Zhu, G. Li, Z. Cui, H. Huang, J. Li, E. Chen, and G. Xu. An Efficient Wikipedia Semantic Matching Approach to Text Document Classification. *Information Sciences*, 393:15 – 28, 2017.

[536] M. Wygralak. Rough Sets and Fuzzy Sets Some Remarks on Interrelations. *Fuzzy Sets and Systems*, 29(2):241–243, 1989.

[537] K. Xiao, S. H. Ho, and A. E. Hassanien. Automatic Unsupervised Segmentation Methods for MRI Based on Modified Fuzzy C-Means. *Fundamenta Informaticae*, 87(3-4):465–481, 2008.

[538] B. K. Xie and S. J. Lee. An Extended Type-Reduction Method for General Type-2 Fuzzy Sets. *IEEE Transactions on Fuzzy Systems*, 25(3):715–724, June 2017.

[539] X. L. Xie and G. Beni. A Validity Measure for Fuzzy Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):841–847, 1991.

[540] Y. Xu, Z. Wu, J. Li, A. Plaza, and Z. Wei. Anomaly Detection in Hyperspectral Images Based on Low-Rank and Sparse Representation. *IEEE Transactions on Geoscience and Remote Sensing*, 54(4):1990–2000, April 2016.

[541] R. R. Yager. A Characterization of the Extension Principle. *Fuzzy Sets and Systems*, 18(3):205–217, 1986.

[542] M. E. Yahia, R. Mahmod, N. Sulaiman, and F. Ahmad. Rough Neural Expert Systems. *Expert Systems with Applications*, 18(2):87 – 99, 2000.

[543] B. Yang, Z. Song, Y. Wang, and W. Song. A New Web Text Clustering Algorithm Based on DFSSM. In *Proceedings of the International Symposium on Electronic Commerce and Security (ISECS '08)*, pages 27–32, Guangzhou City, China, August 2008.

[544] J. Yang, R. Xu, Z. Ding, and H. Lv. 3D Character Recognition using Binocular Camera for Medical Assist. *Neurocomputing*, 220:17 – 22, 2017.

[545] M. Yang and P. Yang. A Novel Condensing Tree Structure for Rough Set Feature Selection. *Neurocomputing*, 71(46):1092 – 1100, 2008.

[546] M.-S. Yang and Y. Nataliani. Robust-Learning Fuzzy C-Means Clustering Algorithm with Unknown Number of Clusters. *Pattern Recognition*, 71:45 – 59, 2017.

[547] P. Yao. Feature Selection Based on SVM for Credit Scoring. In *Proceedings of the International Conference on Computational Intelligence and Natural Computing (CINC '09)*, volume 2, pages 44–47, Wuhan, China, June 2009.

[548] Y. Y. Yao. A Comparative Study of Fuzzy Sets and Rough Sets. *Information Sciences*, 109(1–4):21–47, 1998.

[549] Z. Yao and Q. Zhang. Item-Based Clustering Collaborative Filtering Algorithm under High-Dimensional Sparse Data. In *Proceedings of the International Joint Conference on Computational Sciences and Optimization (CSO '09)*, pages 787–790, Hainan, China, April 2009.

[550] J. Yin and J. Wang. A Model-based Approach for Text Clustering with Outlier Detection. In *Proceedings of the IEEE 32nd International Conference on Data Engineering (ICDE '16)*, pages 625–636, Helsinki, Finland, May 2016.

[551] C. K. Yogesh, M. Hariharan, R. Ngadiran, A. H. Adom, S. Yaacob, and K. Polat. Hybrid BBO_PSO and Higher Order Spectral Features for Emotion and Stress Recognition From Natural Speech. *Applied Soft Computing*, 56:217 – 232, 2017.

[552] A. Yoneyama and T. Minamoto. No-Reference Image Blur Assessment in the DWT Domain and Blurred Image Classification. In *Proceedings of the 12th International Conference on Information Technology - New Generations (ITNG '15)*, pages 329–334, Las Vegas, USA, April 2015.

[553] K. Yu, X. Wu, W. Ding, and J. Pei. Towards Scalable and Accurate Online Feature Selection for Big Data. In *Proceedings of the IEEE International Conference on Data Mining (ICDM '14)*, pages 660–669, December 2014.

[554] L. A. Zadeh. Fuzzy Sets. *Information and Control*, 8:338–353, 1965.

[555] L. A. Zadeh. The Concept of a Linguistic Variable and Its Application to Approximate Reasoning-1. *Information Sciences*, 8:199–249, 1975.

[556] L. A. Zadeh. *Fuzzy Sets and Information Granularity*. North-Holland Publishing Co., Amsterdam, 1979.

[557] L. A. Zadeh. Toward a Theory of Fuzzy Information Granulation and Its Centrality in Human Reasoning and Fuzzy Logic. *Fuzzy Sets and Systems*, 90:111–127, 1997.

[558] R. D. Zarro and M. A. Anwer. Recognition-based Online Kurdish Character Recognition using Hidden Markov Model and Harmony Search. *Engineering Science and Technology, An International Journal*, 20(2):783 – 794, 2017.

[559] Z. Zeng and S. Zhang. Image Classification using Weighted Sparsity Induced Neighbors and Label Embeddings Learning. In *Proceedings of the IEEE International Conference on Electro-Information Technology (EIT '13)*, pages 1–4, Rapid City, USA, May 2013.

[560] D.-Q. Zhang and S.-C. Chen. Kernel Based Fuzzy and Possibilistic C-Means Clustering. In *Proceedings of the International Conference on Artificial Neural Network (ICANN '03)*, pages 122–125, Istanbul, Turkey, June 2003.

[561] H. Zhang, C. Guan, and Y. Li. Signal Processing for Brain-Computer Interface: Enhance Feature Extraction and Classification. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 1615–1618, May 2006.

[562] J. Zhang and L. Gruenwald. An Efficient Optimal Leaf Ordering for Hierarchical Clustering in Microarray Gene Expression Data Analysis. In *Proceedings of the 15th International Workshop on Database and Expert Systems Applications (DExA '04)*, pages 396–400, Zaragoza, Spain, August 2004.

[563] K. Zhang, L. Zhang, and M. H. Yang. Real-Time Object Tracking Via Online Discriminative Feature Selection. *IEEE Transactions on Image Processing*, 22(12):4664–4677, 2013.

[564] M. Zhang, L. D. Xu, W. X. Zhang, and H.-Z. Li. A Rough Set Approach to Knowledge Reduction based on Inclusion Degree and Evidence Reasoning Theory. *Expert Systems*, 20(5):298–304, 2003.

[565] M. Zhang and J. T. Yao. A Rough Sets based Approach to Feature Selection. In *Proceedings of the IEEE Annual Meeting of the Fuzzy Information Processing (NAFIPS '04)*, volume 1, pages 434–439, Banff, Canada, June 2004.

[566] P. Zhang, T. Zhuo, Y. Zhang, L. Xie, and D. Tao. Real-Time Tracking-By-Learning with High-Order Regularization Fusion for Big Video Abstraction. *Signal Processing*, 124:246 – 258, 2016.

[567] Q. Zhang. A Grid Based Clustering Algorithm. In *Proceedings of the 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM '10)*, pages 1–4, Chengdu City, China, 2010.

[568] R. Zhang, J. Shen, F. Wei, X. Li, and A. K. Sangaiah. Medical Image Classification based on Multi-Scale Non-Negative Sparse Coding. *Artificial Intelligence in Medicine*, pages 1 – 10, 2017.

[569] X. Zhang and X. Zhang. Adaptive Multiclass Support Vector Machine for Multimodal Data Analysis. *Pattern Recognition*, 70:177 – 184, 2017.

[570] Z. Zhang and W. Tang. Analysis of Spatial Patterns of Public Attention on Housing Prices in Chinese Cities: A Web Search Engine Approach. *Applied Geography*, 70:68 – 81, 2016.

[571] S. Zhao, E. C. C. Tsang, D. Chen, and X. Wang. Building a Rule-Based Classifier: A Fuzzy-Rough Set Approach. *IEEE Transactions on Knowledge and Data Engineering*, 22(5):624–638, 2010.

[572] X. Zhao, J. Liang, and C. Dang. Clustering Ensemble Selection for Categorical Data based on Internal Validity Indices. *Pattern Recognition*, 69:150 – 168, 2017.

[573] Y. Zhao and H. Lin. WEB Data Mining Applications in E-commerce. In *Proceedings of the 9th International Conference on Computer Science Education (ICCSE '14)*, pages 557–559, Vancouver, Canada, August 2014.

[574] Y. Zheng, Z. Jiang, F. Xie, H. Zhang, Y. Ma, H. Shi, and Y. Zhao. Feature Extraction from Histopathological Images based on Nucleus-Guided Convolutional Neural Network for Breast Lesion Classification. *Pattern Recognition*, 71:14 – 25, 2017.

[575] Z. Zhiwei, G. Xiaomin, W. Lin, Z. Mengmeng, L. Zhikang, Z. Yamin, and Z. Jianhua. Research on Search Engine of Knowledge Adaptation System based on Large Scale Data Set. *Procedia Engineering*, 174:308 – 316, 2017.

[576] N. Zhong and J. Dong. Using Rough Sets with Heuristics for Feature Selection. *Journal of Intelligent Information Systems*, 16(3):199–214, 2001.

[577] N. Zhong, J. Dong, and S. Ohsuga. Using Rough Sets with Heuristics for Feature Selection. *Journal of Intelligent Information Systems*, 16:199–214, 2001.

[578] N. Zhong and A. Skowron. A Rough Set-Based Knowledge Discovery Process. *International Journal of Applied Mathematical Computer Science*, 11(3):603–619, 2001.

[579] M. Zhou, M. Feng, L.-L. Fu, L.-D. Ji, J.-S. Zhao, and J. Xu. Toxicogenomic Analysis Identifies the Apoptotic Pathway as the Main Cause of Hepatotoxicity Induced by Tributyltin. *Food and Chemical Toxicology*, 97:316 – 326, 2016.

[580] W. Zhu and F.-Y. Wang. Reduction and Axiomization of Covering Generalized Rough Sets. *Information Sciences*, 152:217–230, 2003.