Fuzzy-rough discriminative feature selection and classification algorithm, with application to microarray and image datasets

Pramod Kumar P, Prahlad Vadakkepat, and Loh Ai Poh

Department of Electrical and Computer Engineering, 4 Engineering Drive 3, National University of Singapore, Singapore - 117576 (e-mail : prahlad@ieee.org)

Abstract

A novel algorithm based on fuzzy-rough sets is proposed for the feature selection and classification of datasets with multiple features, with less computational efforts. The algorithm translates each quantitative value of a feature into fuzzy sets of linguistic terms using membership functions and, identifies the discriminative features. The membership functions are formed by partitioning the feature space into fuzzy equivalence classes, using feature cluster centers identified by the subtractive clustering technique. The lower and upper approximations of the fuzzy equivalence classes are obtained and the discriminative features in the dataset are selected. Classification rules are generated using the fuzzy membership values that partition the lower and upper approximations. The classification is done through a voting process. Both the feature selection and classification algorithms have polynomial time complexity. The algorithm is tested in two types of classification problems namely cancer classification and image pattern classification. The large number of gene expression profiles and relatively small number of available samples make the feature selection a key step in microarray based cancer

classification. The proposed algorithm identified the relevant features (predictive genes in the case of cancer data) and provided good classification accuracy, at a less computational cost, with good *margin of classification*. A comparison of the performance of the proposed classifier with relevant classification methods shows its better discriminative power.

Keywords: Classifier, Fuzzy-rough sets, Feature selection, Discriminative features, Pattern recognition, Cancer classification, Margin classifier.

1. Introduction

Inductive knowledge acquisition is a prime area of research in pattern recognition. Computational intelligence techniques are useful in such machine learning exercises. Fuzzy and rough sets are two computational intelligence tools used for decision making in uncertain situations. The proposed algorithm utilizes a fuzzy-rough set approach for the feature selection and classification of datasets with large number of features. The presence of large number of features makes the classification of multiple feature datasets difficult. The proposed algorithm is simple and effective in such classification problems.

Similar and overlapped features in a dataset make the classification of patterns difficult. Interclass feature overlaps and similarities lead to indiscernibility and vagueness. Rough set theory [1, 2] is useful for decision making in situations where indiscernibility is present, and, fuzzy set theory [3] is suitable when vague decision boundaries exist.

A rough set [1, 2] is a formal approximation of a vague concept by a pair of precise concepts, the lower and upper approximations. Fuzzy and

2

rough set theories are considered complementary in that they both deal with uncertainty: vagueness for fuzzy sets and indiscernibility for rough sets [4]. These two theories can be combined to form rough-fuzzy sets or fuzzy-rough sets [5, 4]. Combining the two theories provides the concepts of lower and upper approximations of fuzzy sets by similarity relations, which is useful for addressing classification problems.

The computational expenses in any classification process are sensitive to the number of features used to construct the classifier. An abundance of features increases the size of the search space to be explored, thereby increasing the time needed for classification. The available features in a dataset can be categorized into four. 1) Predictive / relevant: the features which are good in the interclass discrimination, 2) Misleading: the features which affect the classification task negatively, 3) Irrelevant: the features which provide a neutral response to the classifier algorithm, and, 4) Redundant: the features of a class which has other relevant features for the discrimination. The presence of misleading features will reduce the classification accuracy and, the presence of irrelevant and redundant features will increase the computational burden. The removal of such features reduces the size of the rule base of a classifier by preserving the relevant and predictive features. This process is known as attribute reduction [6] or, in the context of machine learning, feature selection [7].

This work considers the category of decision problems which is characterized with multidimensional feature space. Searching for an optimal feature subset in a high dimensional feature space is an NP complete problem [8]. In the present work, the feature space is partitioned into fuzzy equivalence

3

classes through fuzzy-discretization. The lower and upper approximations of the fuzzy equivalence classes are calculated for the training data set. The predictive features in the dataset are identified and if-then classification rules are generated using these features. The decision is made through a voting process using these rules. The proposed feature selection and classification algorithm has a polynomial time complexity. The algorithm selects the relevant features, and avoids the misleading, irrelevant and redundant ones. The selection of relevant features reduces the number of features required for classification, which further brings down the computational cost of the classifier.

Cancer and tumor classification using gene expression data is a typical multi-feature classification problem. Gene expression monitoring by DNA microarrays suggests a general strategy for predicting cancer classes, independent of previous biological knowledge [9]. However, the number of genes in the gene expression data is quite large (each gene profile is a feature utilized for the classification) and the availability of tissue samples / records is limited [10]. As the dataset has many more features than the number of available samples, the common statistical procedures such as global feature selection can lead to false discoveries [10]. These facts emphasize the need for a simple and robust classifier for such multi-feature classification problems.

According to Piatetsky-Shapiro *et al.* [10], the main types of data analysis needed for biomedical applications include, *a*) Classification: classifying diseases or predicting treatment outcome based on gene expression patterns, *b*) Gene selection: selecting the predictive features, and, *c*) Clustering: finding new biological classes or refining the existing ones. The first two are

pattern recognition / data mining problems whereas the third requires domain knowledge in the biomedical field. The present work addresses the first two aspects. The proposed feature selection and classification algorithm is applied to five cancer datasets. The algorithm identified predictive genes and provided good classification accuracy for all the datasets considered.

In order to prove the generality of the classifier, the algorithm is also tested on three image datasets. The problems considered are hand posture recognition, human face recognition, and object recognition. The features of the images are extracted using a cortex-like mechanism [11], and, the classification is done using the proposed algorithm.

The structure of this paper is as follows: In Section 2, the concept of fuzzy-rough sets is briefly reviewed, and the related literature is surveyed. Section 3 explains the feature selection and classification algorithm. The algorithm is tested on different datasets (leukemia, multiple tumor, lung cancer, small round blue cell tumor, central nervous system embryonal tumor, hand posture, human face, and object), and the results are discussed in Section 4. Section 5 concludes the paper.

2. Background

2.1. Fuzzy-rough sets

The concept of crisp equivalence class is the basis for rough set theory. A crisp equivalence class contains samples from different output classes. In addition, the various elements in an equivalent class may have different degrees of *belongingness* to the output classes. A combination of fuzzy and rough sets, namely fuzzy-rough sets [4, 5], is useful for decision making in such sit-

uations where both vagueness and indiscernibility are present. Fuzzy-rough set is a popular and effective data mining tool for classification problems due to its strength in handling vague and imprecise data [12]. Fuzzy-rough set is a deviation of rough set theory in which the concept of crisp equivalence class is extended using fuzzy set theory to form fuzzy equivalence class [4]. A fuzzy similarity relation replaces an equivalence relation in rough sets to form the fuzzy-rough sets. In fuzzy-rough sets the equivalence class is fuzzy in addition to the fuzziness of the output classes [13].

Let the equivalence classes be in the form of fuzzy clusters $F_1, F_2, ..., F_H$, which are generated by the fuzzy partitioning of the input set X into H number of clusters. Each fuzzy cluster represents an equivalence class and it contains patterns from different output classes. The definite and possible members of the output class are identified using lower and upper approximations [1] of the fuzzy equivalence classes. The description of a fuzzy set C_c (output class) by means of the fuzzy partitions under the form of lower and upper approximations $\underline{C_c}$ and $\overline{C_c}$ is as follows [13] :

$$
\begin{aligned}
\mu_{\underline{C_c}}(F_j) &= \inf_x \{\max(1 - \mu_{F_j}(x), \mu_{C_c}(x))\} \quad \forall x \in X \\
\mu_{\overline{C_c}}(F_j) &= \sup_x \{\min(\mu_{F_j}(x), \mu_{C_c}(x))\} \quad \forall x \in X
\end{aligned}
\tag{1}
$$

The tuple $\langle \overline{C_c}, \underline{C_c} \rangle$ is a fuzzy-rough set. $\mu_{F_j}(x)$ and $\mu_{C_c}(x)$ are fuzzy memberships of the sample x in the fuzzy equivalence class F_j and output class C_c respectively.

2.2. Related work

The concept of fuzzy discretization of feature space for a rough set theoretic classifier is provided in [14]. The merit of fuzzy discretization over crisp discretization in terms of classification accuracy is demonstrated, when overlapping datasets are used. An entropy-based fuzzy-rough approach for extracting classification rules is proposed in [12]. A new fuzzification technique called Modified Minimization Entropy Principle Algorithm (MMEPA) is proposed to construct membership functions corresponding to the fuzzy sets.

The fuzzy-rough uncertainty is exploited in [13] to improve the classification efficiency of the conventional K-nearest neighbor (K-NN) classifier. The algorithm generalizes the conventional and fuzzy K-NN classifier algorithms. Another modification of the K-NN algorithm using fuzzy-rough sets is proposed in [15]. Fuzzy-rough concept is used to remove those training samples which are in the class boundary and overlapping regions. This improves classification accuracy. However the algorithm is applied to only one type of problem, the hand gesture recognition, whereas [13] reported more experimental results. [16] presents a new fuzzy-rough nearest neighbour (FRNN) classification algorithm, as an alternative to the fuzzy-rough ownership function (FRNN-O) approach in [13]. In contrast to [13], the algorithm proposed in [16] utilizes the nearest neighbors to construct the lower and upper approximations of the decision classes. The algorithm classifies test instances based on their membership to the lower and upper approximations. FRNN outperformed both FRNN-O and traditional fuzzy nearest neighbour (FNN) algorithm.

A new concept named as *consistence degree* is proposed in [17] to use as a critical value to reduce redundant attributes in a database. A rule based classifier using a generalized fuzzy-rough set model is proposed. The classifier is effective on noisy data. A comparison between fuzzy-rough classifier and neural network (NN) classifier is provided in [18]. The fuzzy-rough classifier is reported as a better choice, as it needs lesser training time, has transparency, and has lesser dependance on the training data.

A feature selection method with fuzzy-rough approach and ant colony optimization is provided in [19]. Shen *et al.* [20] proposed a classifier that integrates a fuzzy rule induction algorithm with a rough set assisted feature reduction method. The classifier is tested on two problems, the urban water treatment plant problem and algae population estimation. Fuzzy-rough approach is utilized in [21] for decision table reduction. Unlike other feature selection methods, this method reduces the decision table in both vertical and horizontal directions (both the number of features and its dimensionality are reduced).

A robust feature evaluation and selection algorithm, using a new model of fuzzy-rough sets namely soft fuzzy-rough sets, is provided in [22]. This method is more effective in dealing with noisy data. [23] proposed a fuzzy-rough feature selection algorithm, with application to microarray based cancer classification. These works used standard classifiers (KNN, C5.0) for the classification process.

Most of the above techniques are based on pre-defined fuzzy membership functions and are focussed only on either classification or feature selection aspect. The fuzzy-rough approach is utilized as a preprocessing / supporting

step (except in [12, 17, 21]). Direct construction of classifiers as an application of fuzzy-rough sets has been less studied [17]. The current work proposes a computationally efficient feature selection and classification algorithm for datasets with multidimensional feature space, utilizing the fuzzy-rough set approach. The algorithm automatically generates the fuzzy membership functions and selects the discriminative features in the dataset. The classification rules are generated using the selected subset of features, which further improves the computational efficiency of the classifier.

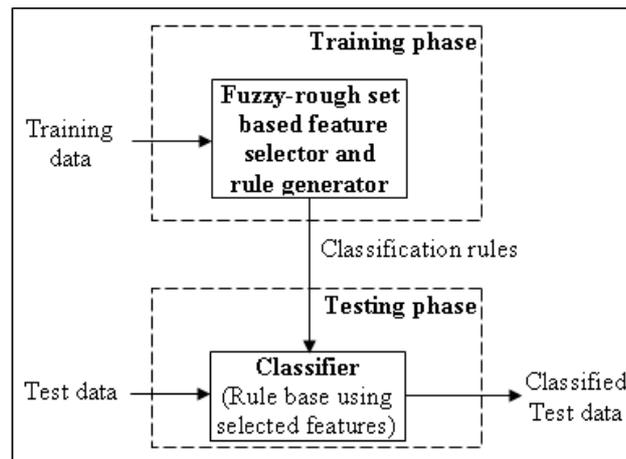## 3. The feature selection and classification algorithm

Figure 1: Overview of the classifier algorithm development.

The proposed fuzzy-rough set based feature selection and classification algorithm is discussed in this section. The aim is to come up with a classifier which identifies discriminative features in a dataset and classifies the data with less computational expense.

Fig. 1 shows the overview of the proposed classifier algorithm development process. The available dataset is divided into two sets: training and testing sets. In the training phase, the discriminative features in the data are selected and the classification rules are generated using a fuzzy-rough approach. The classifier developed is tested using the test data.

3.1. Training phase

The training phase (Fig. 2) involves the fuzzy discretization of the feature space and, the formation of fuzzy membership functions using the cluster centers identified by the subtractive clustering technique [24]. The discriminative features in the dataset are selected and the classification rules are generated, using fuzzy lower and upper approximations of the fuzzified training data.
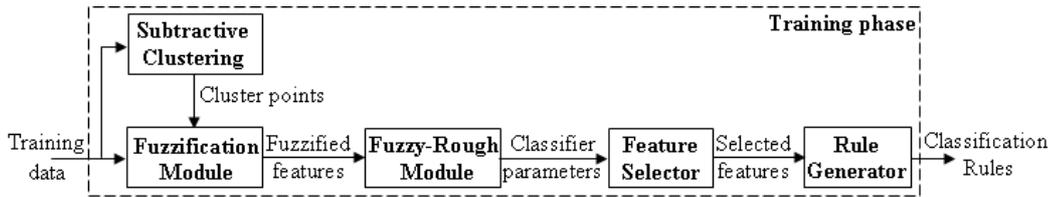

Figure 2: Training phase of the classifier.

3.1.1. Fuzzy equivalence classes and, the lower and upper approximations

The fuzzy membership functions are formed using the feature cluster centers identified by the subtractive clustering technique [24]. Every data point is a potential cluster center. Subtractive clustering calculates a measure of the likelihood of a data point as a cluster center, based on the density of surrounding data points. The algorithm selects the data point with highest potential as the first cluster center and then removes all the data points in

the vicinity (as specified by the subtractive clustering radius which usually lies within $[0.2, 0.5]$) of the first cluster center. The second data cluster and its center point are identified next. This process is repeated until every data sample lies within the radius of one of the cluster centers.

The concept behind the proposed algorithm is explicated with the following example. Consider a 3 class classification problem with a 2 dimensional feature space. Let the sample distribution in the feature space $A - B$ is as shown in Fig. 3(a) and the output class considered is class 2. The fuzzy membership function is centered at the cluster center of feature A (of class 2). The minimum and maximum values of feature A (of class 2) position the left and right sides of the membership function.

In the example, the fuzzy membership function forms an equivalence class. The samples near to the cluster center have maximum membership in class 2, as these samples have a better chance to be in class 2. However the same fuzzy equivalence class contains samples from different output classes which leads to fuzzy-rough uncertainty. The proposed algorithm identifies the membership values μ_{A_L} and μ_{A_H} (4) that partition the definite and possible members of the output class (class 2 in this example, Fig. 3(a)) and identifies the relevant features for discriminating the output class (Section 3.1.2).

Let X be the set of samples labeled 1-8 in Fig. 3(a). The set contains samples from different partitions of the feature space. The lower and upper approximations of the set X are shown in Fig. 3(b). The lower approximation consists the definite members and the upper approximation consists the definite and possible members of class 2.

The cluster centers are identified and the membership functions are formed
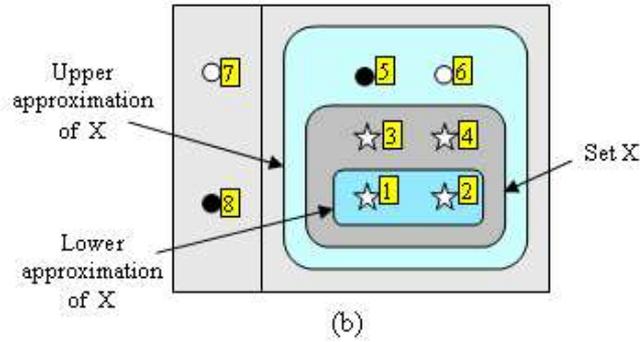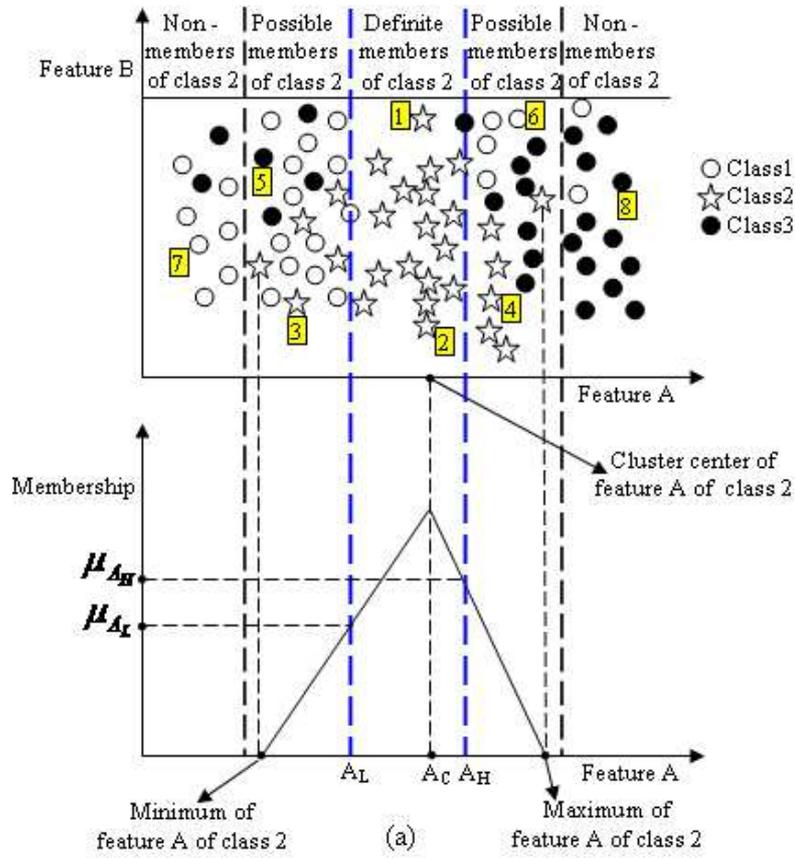
11

Figure 3: (a) Feature partitioning and formation of membership functions from cluster center points in the case of a 3 class dataset. The output class considered is class 2. (b) Lower and upper approximations of the set X which contains samples 1-8 in (a).

for each feature of every class. The training data is then fuzzified using the generated membership functions and, the lower and upper approximations are obtained. The set of membership values $\{\mu_{A_L}, \mu_{A_H}\}$ and feature values $\{A_L, A_H\}$ that partition the definite and possible members are utilized to identify the discriminative features, and to generate the classification rules respectively[1].

The distribution of data samples in the feature space may have varying amount of overlaps between different classes, along various feature axes. The proposed algorithm identifies the discriminative features, the features which have minimum interclass overlap, as the predictive/ relevant features in the dataset, and generates the classification rules using them.

All the datasets considered in this paper have large number of features. The presence of large number of features increases the possibility of identifying predictive features with less interclass overlap.

3.1.2. Feature selection

The fuzzy equivalence class can contain samples from different output classes. Let μ_{A_L} and μ_{A_H} are the membership values that partition the definite and possible members of an output class (Fig. 3(a)). μ_{A_L} and μ_{A_H} are calculated as follows.

Let MF be the fuzzy set associated with a particular feature in a class and,

$$\mu^{max}(C_i) = \max\{\mu_{MF}[A_{C_i}(l)]\}, \tag{2}$$

---

[1]Refer Appendix-A for an illustration of the formation of fuzzy membership functions, and the calculation of $\{\mu_{A_L}, \mu_{A_H}\}$ and $\{A_L, A_H\}$.

where,

$A_{C_i}(l)$ represents the feature values of the samples from the class C_i,

and,

$$C_{max} = \text{argmax}_{C_i}\{\mu^{max}(C_i)\}, \tag{3}$$

then,

$$\mu_{A_L} = \max_{C_i \neq C_{max}, A_{min} \leq A \leq A_C} \{\mu^{max}(C_i)\},$$
$$\mu_{A_H} = \max_{C_i \neq C_{max}, A_C \leq A \leq A_{max}} \{\mu^{max}(C_i)\}. \tag{4}$$

μ_{A_L} and μ_{A_H} are the maxima of the membership values associated with data samples belonging to classes other than C_{max}. Once these membership values are calculated, the features of the data are sorted in descending order of d_μ where d_μ is the average value[2] of $d_{\mu_{A_L}}$ and $d_{\mu_{A_H}}$ (Fig. 4).

$$d_\mu = (d_{\mu_{A_L}} + d_{\mu_{A_H}})/2, \tag{5}$$

where,

$$d_{\mu_{A_L}} = 1 - \mu_{A_L} \text{ and } d_{\mu_{A_H}} = 1 - \mu_{A_H} \tag{6}$$

The classification rules are generated using the first n features from the sorted list (using the features which provide higher values of d_μ). The proposed algorithm is tested by varying the number of selected features n. The classification accuracy of the algorithm first increases and then saturates with respect to n, for all the datasets considered.

---

[2]The algorithm provides better results when the average value is considered. The other possible values for d_μ are $d_{\mu_{A_L}}$, $d_{\mu_{A_H}}$ or a weighted average of $d_{\mu_{A_L}}$ and $d_{\mu_{A_H}}$.
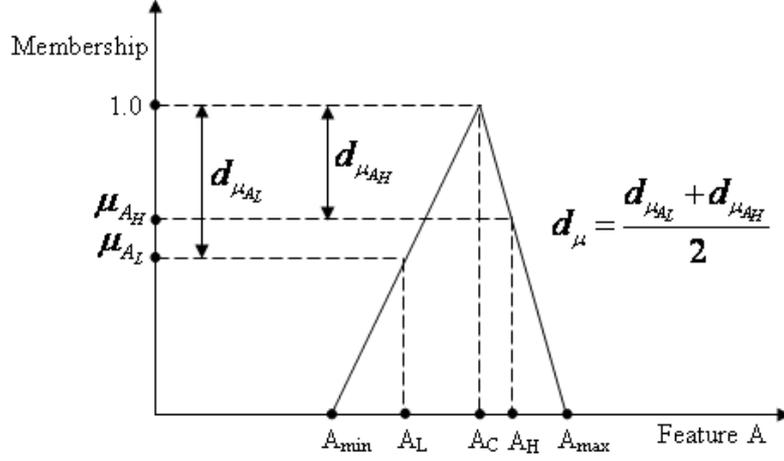
Figure 4: Calculation of d_μ.

The value of d_μ is an indication of the discriminative ability of a particular feature. A high value of d_μ indicates that the corresponding feature is good for interclass discrimination, as it has less interclass overlap. $d_\mu = 0$ ($\mu_{A_L} = \mu_{A_H} = 1$) represents an indiscriminating feature.

The capability of d_μ in representing the relevance of features is explained as follows. The calculation of d_μ for two features A1 and A2 is depicted in Fig. 5. The range of feature A1 ($A1_{max} - A1_{min}$) is less than that of A2 ($A2_{max} - A2_{min}$). Let the feature range $[A_L, A_H]$ and feature cluster center A_C are the same for the two features A1 and A2. In this case the feature selection algorithm provides preference to feature A1, as it has a denser sample distribution. The algorithm provides higher value for $d_{\mu_{A1}}$ (the average of $d_{\mu_{A1_L}}$ and $d_{\mu_{A1_H}}$) than $d_{\mu_{A2}}$. Within the feature range of A1, $A1_{max} - A_C < A_C - A1_{min}$, which implies that the distribution of samples is sparser within $[A1_{min}, A_C]$ and denser within $[A_C, A1_{max}]$ (as A_C is the
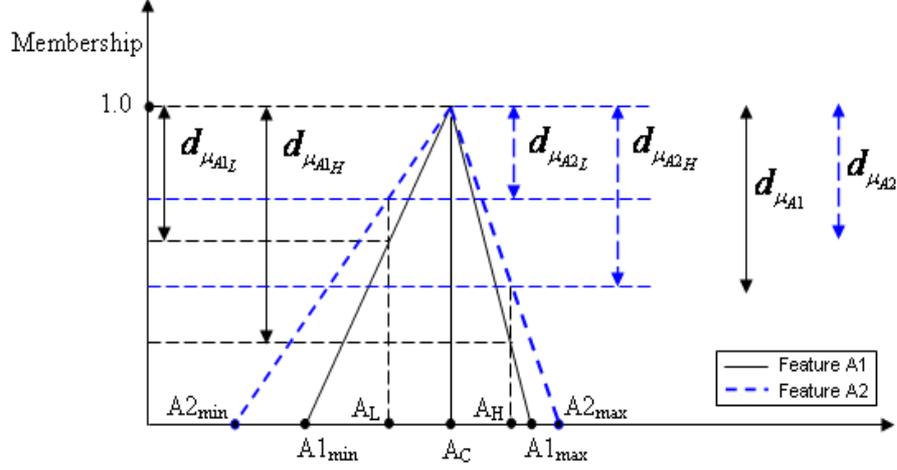
15

Figure 5: Calculation and comparison of d_μ for two features A1 and A2 with different feature ranges.

feature cluster center). The proposed algorithm provides preference to the denser feature range $[A_C, A_H]$ and assigns a high value to the d_μ corresponding to it $(d_{\mu_{A1_H}})$, compared to the d_μ of the sparse feature range $[A_L, A_C]$ $(d_{\mu_{A1_L}})$.

3.1.3. Classification

The feature values A_L and A_H that partition the lower and upper approximations of the fuzzy equivalence class entails the rule: if the value of a feature A is within (A_L, A_H) then the sample belongs to the class C_{max} (Fig. 4, A.11). This feature range decides whether a particular sample is a definite member of the output class C_{max}. The rule always holds true for the training samples. However some of the rules classify only a small number of training samples (say 1 or 2), if the samples from various classes are well mixed. To increase the reliability of the classifier, only those rules which classify two or

more number of training samples are stored in the rule base.

In order to classify new samples, the classification rules are generalized as follows (*Rule 1 & 2*).

Let $\{A_{L_{ij}}, A_{H_{ij}}\}$ be the set of feature values obtained as per (4) where,

i $1,\ldots, p$: p - number of classes,

j $1,\ldots, q'$: q' - number of selected features,

then the samples are classified using the following two rules. *Rule 1* is a voting step whereas *Rule 2* is the decision maker.

Rule 1:

$$IF \ [A_{L_{ij}} < A_j < A_{H_{ij}}] \ THEN \ [N_{C_i} = N_{C_i} + 1] \tag{7}$$

where,

A_j j^{th} feature of the sample to be classified,

N_{C_i} the number of votes for a particular class C_i.

Rule 2:

$$C = argmax_{C_i}\{N_{C_i}\} \tag{8}$$

where,

C is the class to which the sample belongs (the output class).

A detailed flowchart of the training phase of the classifier is provided in Fig. 6.

The proposed classifier is a *margin classifier* that provides the minimum distance from the classification boundary, namely *margin of classification (MC)*, for each sample. The margin of classification for the proposed classifier
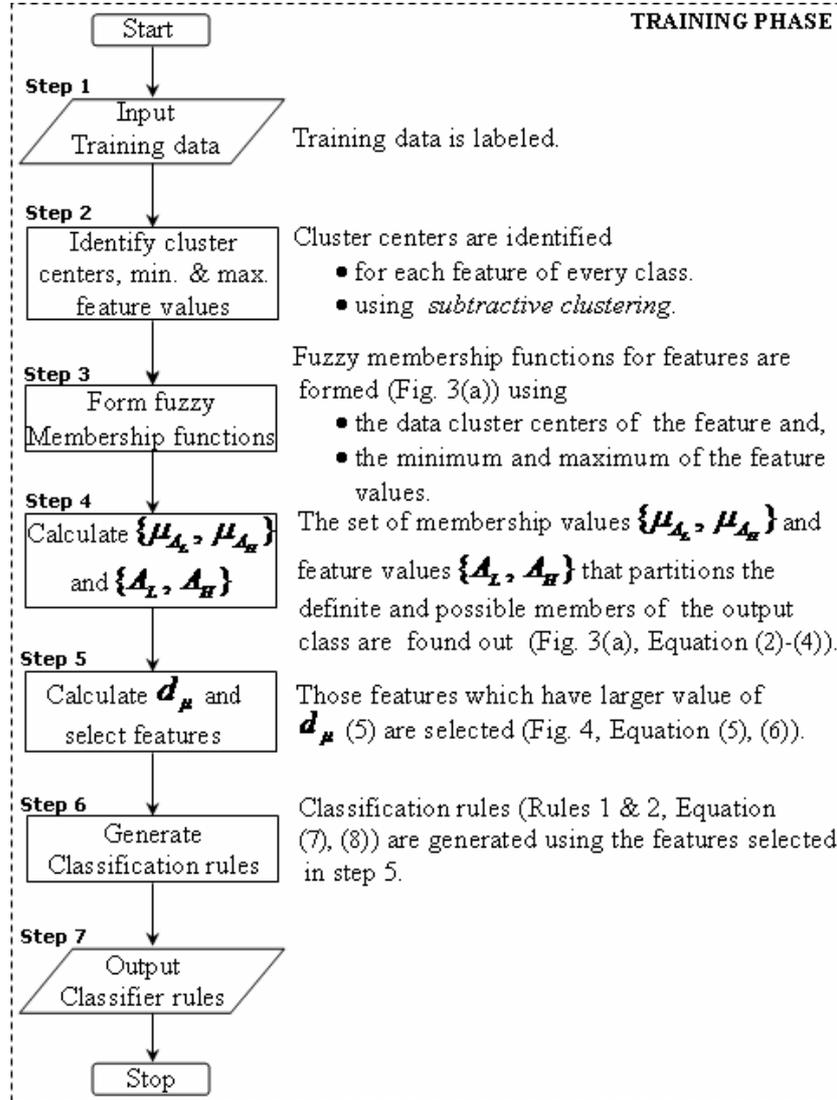
TRAINING PHASE

Start

Step 1
Input Training data
Training data is labeled.

Step 2
Identify cluster centers, min. & max. feature values

Cluster centers are identified
- for each feature of every class.
- using *subtractive clustering*.

Step 3
Form fuzzy Membership functions

Fuzzy membership functions for features are formed (Fig. 3(a)) using
- the data cluster centers of the feature and,
- the minimum and maximum of the feature values.

Step 4
Calculate $\{\mu_{A_L}, \mu_{A_H}\}$ and $\{A_L, A_H\}$

The set of membership values $\{\mu_{A_L}, \mu_{A_H}\}$ and feature values $\{A_L, A_H\}$ that partitions the definite and possible members of the output class are found out (Fig. 3(a), Equation (2)-(4)).

Step 5
Calculate d_μ and select features

Those features which have larger value of d_μ (5) are selected (Fig. 4, Equation (5), (6)).

Step 6
Generate Classification rules

Classification rules (Rules 1 & 2, Equation (7), (8)) are generated using the features selected in step 5.

Step 7
Output Classifier rules

Stop

Figure 6: Flowchart of the training phase.

18

is defined as,

$$MC = number\ of\ positive\ votes - max.(number\ of\ negative\ votes). \quad (9)$$

In case of a dataset with three classes, the number of votes N_{C_1}, N_{C_2}, and N_{C_3} are calculated for a sample under consideration (*Rule 1*). *Rule 2* then identifies the class with maximum votes. *Rules 1* and *2* serve to form the classifier rule base, keeping the algorithm computationally simple.

For a sample from class 1, let the values $N_{C_1} = 90$, $N_{C_2} = 5$, and $N_{C_3} = 10$. Then the MC for the sample is $90 - 10 = 80$. In this case the sample received 90 positive votes[3]. The number of negative votes are 5 and 10 in classes 2 and 3 respectively. A positive margin indicates correct classification whereas negative margin indicates misclassification. The average margin of classification for a dataset indicates the discriminative power of the classifier for that dataset. The experimental results (Section 4) evidence good discriminative power of the proposed classifier for all the datasets considered.

3.2. Testing phase

The fuzzy-rough classifier is formed using the classification rules generated in the training phase. Fig. 7 shows the flowchart of the testing phase (the classifier). The selected features of a test sample are compared with the feature values A_L and A_H, and the classification is done using *Rules 1* and *2*. Each of the execution steps of the classifier is a direct comparison

---

[3]Voting is positive if the voted class and the actual class are the same. Otherwise it is negative.

of feature values, using the classification rules, which makes the algorithm computationally simple. The classification results are discussed in Section 4.
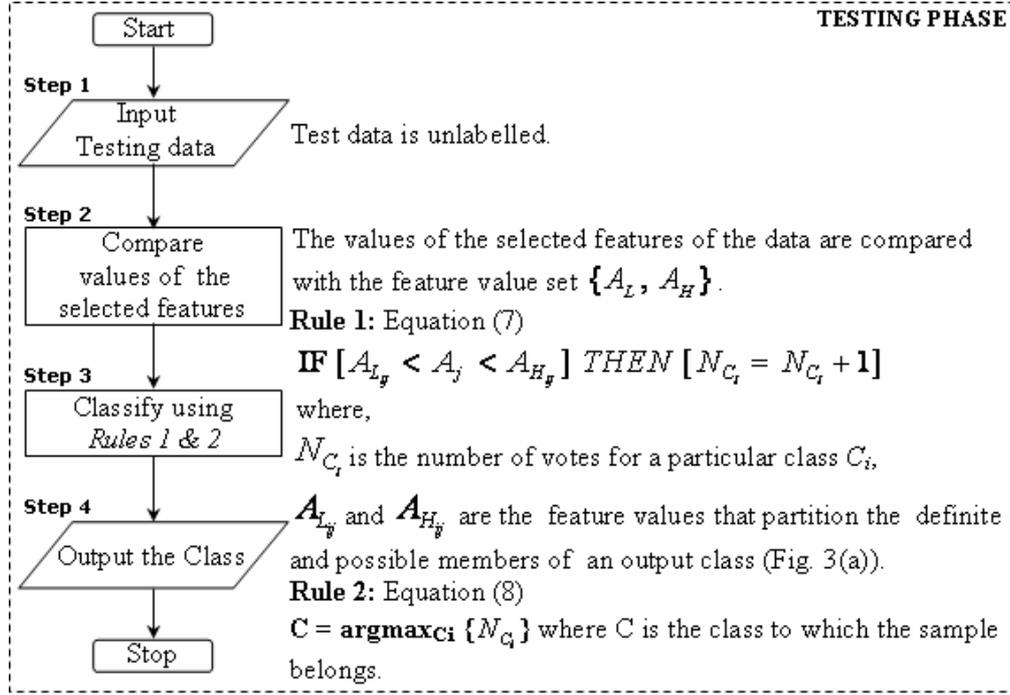

Figure 7: Flowchart of the testing phase.

3.3. Computational complexity analysis

This section provides a detailed computational complexity analysis of the training and the testing (the classifier) algorithm. Both training and testing algorithms have polynomial time complexity.

3.3.1. Computational complexity of the classifier training algorithm

Fig. 8 shows the pseudo code of the classifier training algorithm. The different parameters at the input of the training algorithm are the number of classes, the number of features, and the number of samples per class. Let p be the number of classes, q be the number of features, and r be the number of samples per class. The complexity of the algorithm is as follows : $O(pqr)$ for reading the training data, $O(pqr^2)$ for finding the cluster centers (using subtractive clustering), $O(pqr)$ for finding the minimum and maximum of feature values, $O(p^2qr)$ for calculating the fuzzy memberships, $O(p^2qr)$ for finding the membership values μ_{A_L} and μ_{A_H}, $O(pq)$ for finding feature values A_L and A_H, $O(pq)$ for calculating d_μ, $O(pq\ log(q))$ for sorting d_μ, and, $O(pq)$ for storing the rule base parameters. The overall complexity of the algorithm is $O(pqr^2) + O(p^2qr) + O(pq\ log(q))$, which is polynomial time.

3.3.2. Computational complexity of the classifier

The pseudo code of the classifier is shown in Fig. 9. The different parameters at the input of the classifier are the number of classes and the number of selected features. Let p be the number of classes and q' be the number of selected features. The complexity of the algorithm is as follows : $O(pq')$ for reading the classifier rule parameters, $O(q')$ for reading the selected features of the sample, $O(pq')$ for the voting process, and, $O(p)$ for finding the class index which received the maximum votes. The overall complexity of the proposed classifier algorithm is polynomial time, $O(pq')$.

```
read {train_data} ; % read all the features of the samples
for i = number_of_classes
   for j = number_of_features;
      calculate {A_c} ; % finding feature cluster centers - using subtractive clustering
      calculate {A_min}, {A_max}; % finding min. & max. values of features
      calculate {μ}; % calculation of fuzzy memberships in each class
      calculate {μ_{A_L}, μ_{A_H}}; % Equation (2)-(4)
      calculate {A_L, A_H};  % finding features values  that partitions definite
                        % and possible members of an output class
      calculate {d_μ};  % Equation (5), (6)
   end
end
for i = number_of_classes
      A_sort = sort {d_μ}; % features are sorted in the descending order of d_μ
                     % - for the feature selection
end
write {A_L, A_H}, {A_sort}; % store the parameters of the rule base
```

Figure 8: Pseudo code of the classifier training algorithm.

```
read {$A_L$, $A_H$}, {$\mathbf{A_{sort}}$}; % read the parameters of the rule base

read {test_data} ; % read the selected n features of the sample - corresponding
                % to the first n feature indices listed in {$\mathbf{A_{sort}}$}

for i = number_of_classes

   for j = number_of_features; % selected n number of features

      IF  [$A_L < A < A_H$]  % compare the features of the test sample with

                % feature value set {$A_L$, $A_H$}

            vote(i) = vote(i) +1; % Rule 1, Equation (7)

      end

   end

end

class = index {max {vote}} ; % Rule 2, Equation (8),  find the class index which
                % received maximum votes

print {class} ;
```

Figure 9: Pseudo code of the classifier.

4. Performance evaluation and discussion

The proposed feature selection and classification algorithm is tested using 5 cancer (Table 1) and 3 image datasets (Table 2). The reported results are the classification accuracy, the number of features used per class, the total number of features used (which is less than or equal to the product of number of features used per class and the number of classes), and the average margin of classification (MC) (Table 3 & 5). The variation in classification accuracy with respect to number of selected features is reported (Fig. 10). A comparison of the classification accuracy of the proposed algorithm with relevant classification methods is also provided.

Table 1: Details of cancer datasets

Dataset	# Classes	# Samples
Leukemia [25]	3	72
Tumor vs. normal samples [25]	2	75
Lung cancer [26]	2	181
Small round blue cell tumor [27]	4	83
Central nervous system embryonal tumor [25]	5	42

4.1. Cancer classification

Cancer classification, which is a typical multi-feature classification problem, is based on microarray based gene expression data. Accurate classification of cancer is necessary for diagnosis and treatment. As the number

24

Table 2: Details of hand posture, face and object datasets

Dataset	# Classes	# Samples
Jochen Triesch hand posture dataset [28]	10	240
A subset of Yale face dataset [29]	10	640
Caltech object database [30]	4	3479

of available samples is limited, 10 fold cross-validation is done for all cancer datasets, except for the fifth dataset (central nervous system embryonal tumor), for which *leave one out cross validation* is done. The classification results are compared with that of Support Vector Machines (SVMs) implemented using LIBSVM [31] (Table 3). The classification results are also compared with those reported in the literature (for this comparison, the training and testing of the algorithm are done using the same sample divisions as that in the compared work) (Table 4).

4.1.1. Leukemia classification

The leukemia dataset [25] consists of a total of 72 samples. Each sample has 7,129 gene expression profiles and each gene profile is a feature in the classification process. Originally the dataset was created and analyzed for the binary classification into acute lymphoblastic leukemia (ALL) and acute myeloblastic leukemia (AML) [9]. Jirapech-Umpai *et al.* [32] separated the dataset into three classes by using subtypes of ALL. Seventy two samples in the dataset are divided into three classes: ALL B-cell (38), ALL T- cell (9), and AML (25). In the present work, the three class classification is

carried out and the 55 top ranked genes[4] by the RankGene method [32] are utilized for classification. For the 10 fold cross validation, 4 samples of ALL B-cell, 1 sample of ALL T-cell and 3 samples of the AML are considered in one subset (some samples are repeated in the subsets, as 10 subsets each having 8 samples are formed using the 72 samples). The outcome of the classification is provided in Table 3. The proposed algorithm provided the maximum classification accuracy of 100%, when a total of 35 features are used. The variation in classification accuracy with respect to the number of selected features per class is shown in Fig. 10(a). The SVM classifier provided 94.44% accuracy for this dataset (even though all available features are used in SVM).

The classification of the same dataset is done in [32] using evolutionary algorithm and GA-KNN classifier which reported 98.24% accuracy. In their work, 38 samples are used for training and 34 samples are used for testing. The proposed algorithm is tested using the same sample divisions and provided 100% classification accuracy (Table 4). Fifty gene profiles (features) are used in [32], whereas 35 gene profiles (15 selected features per class) are used by the proposed algorithm.

4.1.2. Tumor vs. Normal sample classification

In [33], tumor detection is done using MicroRNA expression profiles. The dataset [25] consists of expression profiles of tumor and normal samples of multiple human cancers. Each sample has 217 expression profiles. A k-Nearest Neighbor (kNN) classifier is built and trained using the human tumor

---

[4]The list of 55 top ranked genes is available in [32].

/ normal samples and it is utilized for the prediction of tumor in mouse lung samples [33]. Also the algorithm identified markers, the features that best distinguishes tumor and normal samples. The training data consists of colon, kidney, prostate, uterus, lung, and breast human tumor / normal samples (43 tumor and 32 normal samples) and the testing is done using mouse lung tumor / normal samples (12 samples). The kNN classifier provided 100% correct detection of tumor (Table 4), when 131 markers are used [33].

As this is a tumor detection problem (rather than a classification problem), the lower and upper approximations, and the corresponding membership (μ_{A_L}, μ_{A_H}) and feature values (A_L, A_H) are calculated only for the tumor samples (and not for the normal samples). The presence of cancerous tumor in the test sample is predicted if more than 50% of the selected features vote the sample as tumor sample. The algorithm provided 100% correct detection of tumor in the mouse lung samples, when 35 markers (20 selected features per class) are used.

In order to substantiate the reliability of the classifier, an additional 10 fold cross test is done using the human tumor / normal samples (43 tumor and 32 normal samples). Each of the subsets consists of 5 tumor and 4 normal samples (some samples are repeated in the subsets, as 10 subsets each having 9 samples are formed using the 75 samples). The proposed algorithm provided 96.67% classification accuracy when a total of 35 selected features are used (Table 3). Fig. 10(a) shows the variation in classification accuracy with respect to the number of selected features per class. For this dataset the SVM classifier provided an accuracy of 94.66%.

4.1.3. Lung cancer classification

Lung cancer dataset [26] is used by Gordon *et al.* [34] for gene expression ratio based diagnosis of mesothelioma (MPM) and adenocarcinoma (ADCA) of the lung. The set contains gene expression profile of 181 tissue samples (31 MPM and 150 ADCA). For the lung cancer dataset the number of genes is 12,533 and all of them are considered for the classification. Three MPM and 15 ADCA samples are used in one subset to do the 10 fold cross validation (One subset contains 18 samples. The execution is repeated until all the 181 samples are tested). The proposed algorithm provided 100% classification accuracy (Table 3) when a total of 44 features (40 selected features per class) are used. The variation in classification accuracy with respect to the number of selected features per class is shown in Fig. 10(b). For this dataset the SVM classifier also provided 100% accuracy.

In [34], 99% classification accuracy is achieved for the same dataset, when 16 MPM and 16 ADCA samples are used for training, whereas the proposed algorithm provided 100% accuracy, when trained in a similar manner (Table 4). However, only 6 genes are used in [34] as the method is based on the ratio of gene expressions, whereas the proposed algorithm needs 24 genes to provide 99% accuracy and 44 genes to provide 100% accuracy.

4.1.4. Small round blue cell tumor classification

The dataset used is NCI's dataset [27] of small round blue cell tumors (SRBCTs) of childhood [35]. There are four classes : Burkitt lymphoma (BL), Ewing sarcoma (EWS), Neuro blastoma (NB) and Rhabdomyosarcoma (RMS). A total of 83 samples are provided (11 BL, 29 EWS, 18 NB and 25 RMS). The dataset consists of 2,308 genes whereas the present work utilizes

the top 200 individually discriminatory genes (IDGs)[5] identified by Xuan *et al.* [36]. For the 10 fold cross test 1 BL, 3 EWS, 2 NB and 2 RMS are considered in one subset (One subset contains 8 samples. The execution is repeated until all the 83 samples are tested). The classification results are provided in Table 3. The proposed algorithm provided an accuracy of 98.75%, when a total of 103 features are used. Fig. 10(b) shows the variation in classification accuracy with respect to the number of selected features per class. For this dataset the SVM provided similar performance as that of the proposed algorithm. The accuracy provided by SVM classifier is 98.8%.

The classification of the same dataset is done in [35], using neural network (NN) classifier with 96 features, and in [36], using multilayer perceptron (MLP) classifier with 9 features. They used 63 samples for training and 20 samples for testing, and achieved 100% and 96.9% classification accuracies respectively. The proposed algorithm provided an accuracy of 100% when the same training and testing samples are used, with 103 features (35 selected features per class). The proposed classifier provided better accuracy compared to the MLP classifier. However, in the MLP classifier, the classification is done with less number of genes (only 9) as Jointly Discriminatory Genes (JDGs) [36] are used.

4.1.5. *Central nervous system embryonal tumor classification*

The gene expression based prediction of central nervous system embryonal tumor is reported in [37]. The multiple tumor classes are predicted using k Nearest Neighbor (kNN) algorithm. There are five classes namely

---

[5]These genes are listed in the additional material (Table S1) of [36].

medulloblastoma (10 samples), malignant glioma (10 samples), AT/RT (an aggressive cancer of the central nervous system, kidney, or liver that occurs in very young children) (10 samples), normal cerebellum (4 samples), and supratentorial PNETs (primitive neuroectodermal tumor) (8 samples). A total of 42 samples are provided, each having 7129 gene profiles. The evaluation method used is *leave one out cross validation*, similar to [37]. The accuracy achieved by the proposed algorithm is 85.71 %, whereas that achieved by the kNN classifier is 83.33% (Table 3). The proposed algorithm used 90 selected genes (markers) per class whereas [37] used only 10 markers per class (the proposed algorithm provided a lesser accuracy when only 10 markers per class are used - Fig. 10(c)). The accuracy provided by the SVM classifier for this dataset is 80.95%.

Table 3: Summary and comparison of cross validation test results - Cancer datasets (Training and testing are done by cross validation)

Dataset	Accuracy (%) -proposed method	# features / class	Total # features	Average MC	Accuracy (%) -SVM
Leukemia	100	15	35	7.03	94.44
Tumor vs. normal samples data	96.67	20	35	-	94.66
Lung cancer	100	40	44	22.44	100
Small round blue cell tumor	98.75	35	103	11	98.80
Central nervous system embryonal tumor	85.71	90	433	35	80.95

Table 4: Comparison of classification accuracy (%) with reported results in the literature
- Cancer datasets (Training and testing are done using the same sample divisions as that
in the compared work)

Dataset	Proposed method	Benchmark
Leukemia	100	98.24 [32]
Tumor vs. normal samples data	100	100 [33]
Lung cancer	100	99 [34]
Small round blue cell tumor	100	100 [35]
Central nervous system embryonal tumor	85.71	83.33 [37]

4.2. Image pattern recognition

The image based pattern classification problems considered are hand posture recognition, face recognition, and object recognition. Image features used are C_2 standard model features (SMFs) [11], which are extracted using a computational model of the ventral stream of visual cortex [11, 38]. A total of 1000 C_2 SMFs are extracted. Two fold cross validation is done for hand posture and face datsets. For the object dataset, 100 random images per class are used for training, and the testing is done using all the remaining images. The reported classification accuracy is the average over 10 such runs. The classification results for image datasets are compared with that of SVM and principal component analysis (PCA-eigenface method [39]), both implemented using the C_2 SMFs (Table 5).

4.2.1. Hand posture recognition

Hand posture dataset considered is Jochen-Triesch hand posture dataset [40]. Ten classes of hand postures, performed with 24 different persons, against light background, are considered for the classification. The images vary in size of the hand and shape of the postures. A total of 240 samples, with 24 images from each class, are provided. The algorithm is tested by dividing the dataset equally into training and testing samples (12 images from each class). The algorithm is repeated by interchanging training and testing sets and the average results are reported (two fold cross test).

The proposed algorithm provided the maximum classification accuracy of 100%, whereas PCA provided an accuracy of 98.75% (Table 5). The SVM classifier also provided 100% accuracy. However, all the 1000 features are used for the classification using PCA and SVM, whereas the proposed algorithm used only 178 features (20 selected features per class). The variation in classification accuracy with the number of selected features per class is shown in Fig. 10(d). The accuracy saturates at 100% when 20 or more selected features per class are used.

4.2.2. Face recognition

Face dataset considered is a subset of the Yale face database B [29], which contains 10 classes of face images, taken from different lighting directions. It consists of 640 frontal face images, 64 from each class. The algorithm is tested in a similar manner as done for the hand posture dataset. The proposed algorithm as well as the SVM classifier provided the maximum classification accuracy of 100%, whereas the PCA provided 99.7% accuracy (Table 5). The whole set of features (1000) is used in PCA and SVM. The

32

proposed algorithm is tested by varying the number of selected features (Fig. 10(d)). The classification accuracy saturates at 100% when only 7 selected features per class (a total of 56 features) are used.

Table 5: Summary and comparison of cross validation test results - hand posture, face and object recognition

Dataset	Accuracy (%) -proposed method	# features / class	Total # features	Average MC	Accuracy (%) -PCA	Accuracy (%) -SVM
Hand posture dataset	100	20	178	12.65	98.75	100
Face dataset	100	7	56	5.68	99.7	100
Object dataset	94.96	8	30	3.39	88.37	94.84

4.2.3. Object recognition

The classes considered in object recognition are human frontal face, motorcycle, rear car, and airplane [30]. A total of 3479 images (450 faces, 800 motor cycles, 1155 cars, and 1074 air-planes) are provided. The training set consists of 100 randomly selected images from each class, and the testing is done using remaining 3079 images. The reported results are the average over ten such runs. The proposed algorithm provided an average classification accuracy of 94.96%, whereas PCA provided 88.37% accuracy (Table 5). The SVM classifier provided an accuracy of 94.84%, equivalent to that of the proposed algorithm. All the 1000 features are used in PCA and SVM, whereas the proposed algorithm needs only 30 features (8 selected features

per class). The recognition accuracy saturates at 94.96% when 8 or more selected features per class are used (Fig. 10(d)).
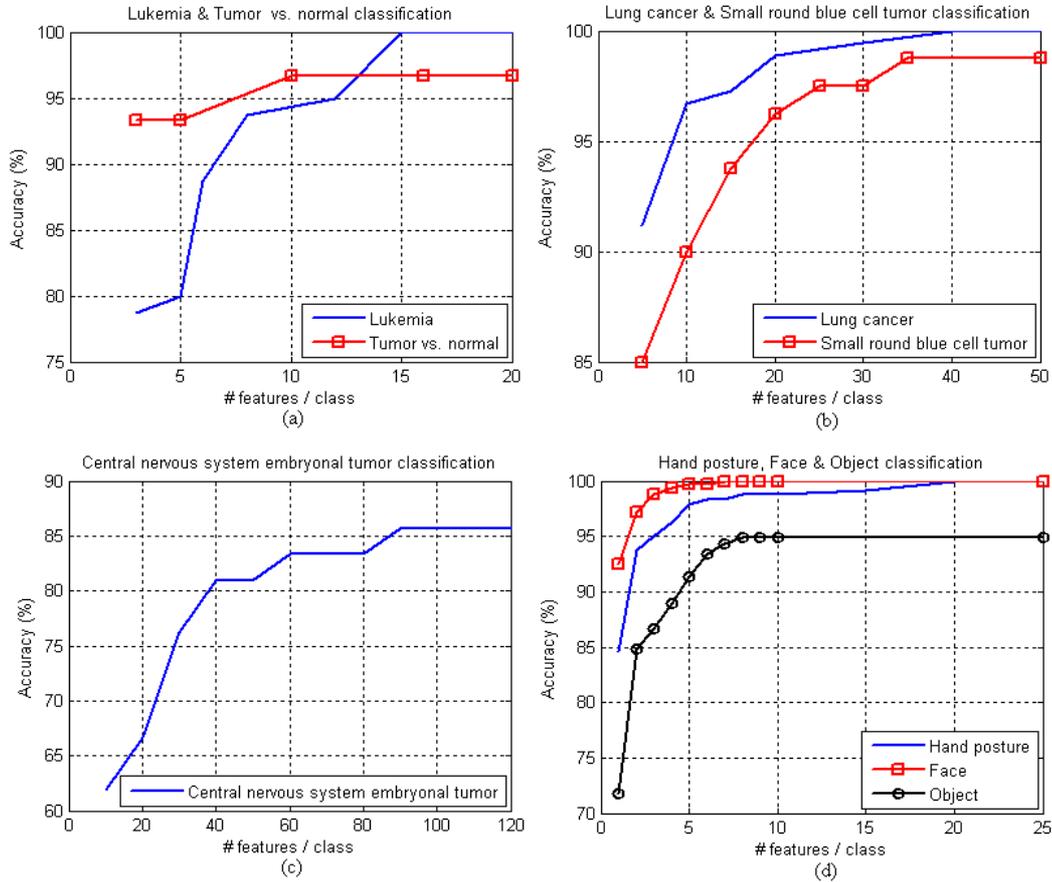

Figure 10: Variation in classification accuracy with the number of selected features.

4.3. Future work

In the present work the first (main) cluster center of a feature is considered for the generation of fuzzy membership functions. More membership functions (and classification rules) can be generated by considering the second and third cluster centers, which can improve the performance of the

algorithm. The feature selection algorithm needs modification accordingly, as the density of cluster centers vary (in general, first cluster center is denser than the second, which in turn is denser than the third).

In the case of image datasets, the identification of discriminative C2 features provides an insight into the functioning of the C2 feature extraction system, which imitates the primate visual cortex. Each image feature corresponds to a prototype patch [11] extracted from the training images. The selection of relevant features identifies the discriminative prototype patches, which enhances the shape selectivity. This information can be utilized to tune the different parameters in the feature extraction system, in order to improve its performance.

5. Conclusion

A feature selection and classification algorithm based on the concept of fuzzy-rough sets is proposed. The fuzzy membership functions, which partition the feature space into fuzzy equivalence classes, are evolved automatically from the training dataset. The fuzzy membership values that partition the lower and upper approximations of the fuzzy equivalence classes are utilized to identify the discriminative features in the dataset. The classifier rules are generated using the identified predictive features and the samples are classified through a voting process using these rules. A measure of the quality of classification, the *margin of classification*, is defined for the proposed classifier.

The performance of the algorithm is evaluated with two types of multiple feature datasets namely cancer and image datasets, and compared with

relevant classification techniques. The proposed algorithm provided classification accuracy which is equivalent or better than that provided by the compared methods, with less computational efforts, and with good margin of classification.

Selection of relevant features reduced the number of features required for classification, which in turn reduced the computational burden of the classifier. The classification accuracy first increases and then saturates with respect to the number of selected features, for all the datasets considered. The effectiveness of the classifier in different types classification problems proves its generality. The time needed for rule generation as well as classification is several seconds, whereas that of the conventional machine learning algorithms is of the order of minutes or even hours. The proposed classifier is effective in cancer and tumor classification, which has high impact in the biomedical field. Also the effectiveness of the algorithm in image pattern recognition is evident, which is useful in human-computer interaction, human-robot interaction, and virtual reality.

Appendix A. Illustration of the formation of fuzzy membership functions, and the calculation of $\{\mu_{A_L}, \mu_{A_H}\}$ and $\{A_L, A_H\}$ - Object dataset

Fig. A.11(a)-(d) shows the identified best discriminative feature for a particular class, which has a well separated feature cluster center (the center of the fuzzy membership function). The selection of such features ease the classification process, even though there is an interclass feature overlap. Learning classification rules from the features which have distribution sim-
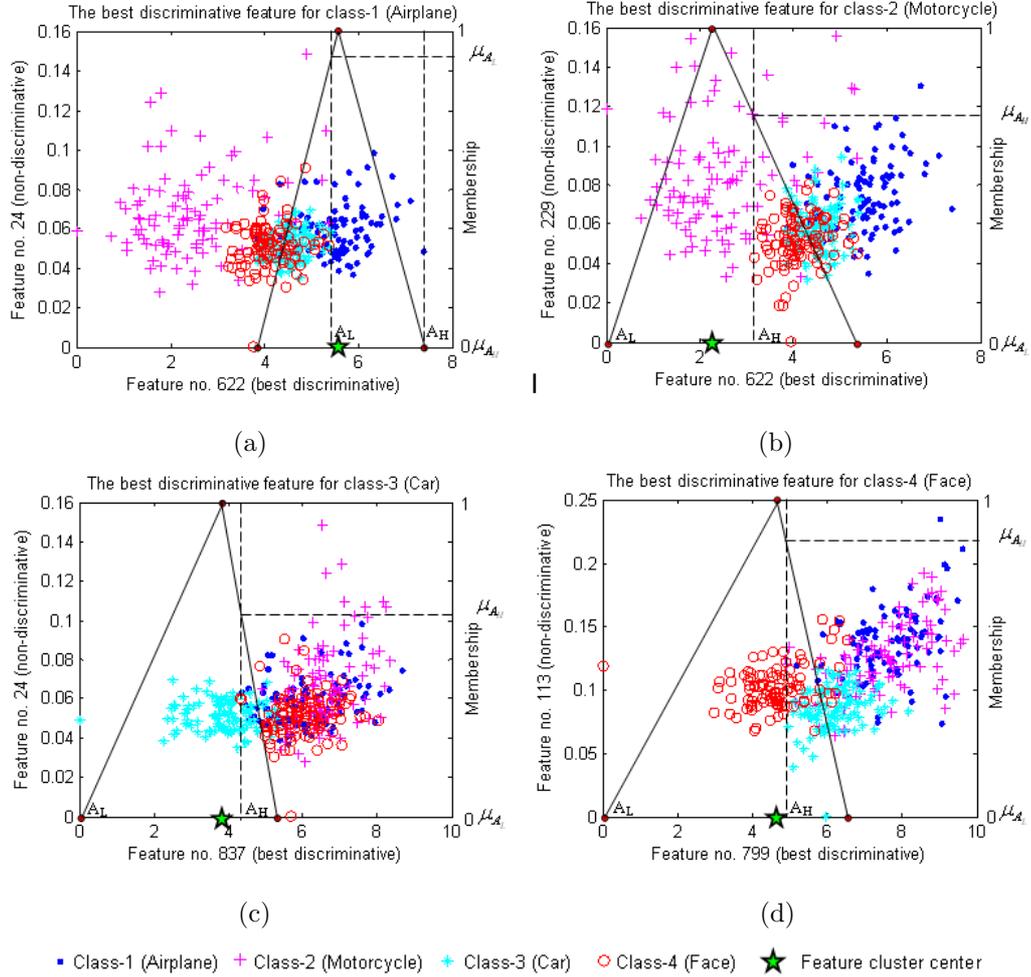
Figure A.11: Illustration of the formation of fuzzy membership functions, and the calculation of $\{\mu_{A_L}, \mu_{A_H}\}$ and $\{A_L, A_H\}$ for object dataset. Subfigures (a)-(d) show a two dimensional distribution (only two feature axes are shown) of training samples in the object dataset (class 1-4 respectively). The x-axis represents the best discriminative feature (which is selected) and the y-axis represents one of the non-discriminative features (which are not selected). Subfigures (a)-(d) also show the formation of fuzzy membership functions, the calculation of the membership values $\{\mu_{A_L}, \mu_{A_H}\}$ and the feature values $\{A_L, A_H\}$ (Section 3.1), for the four object classes.
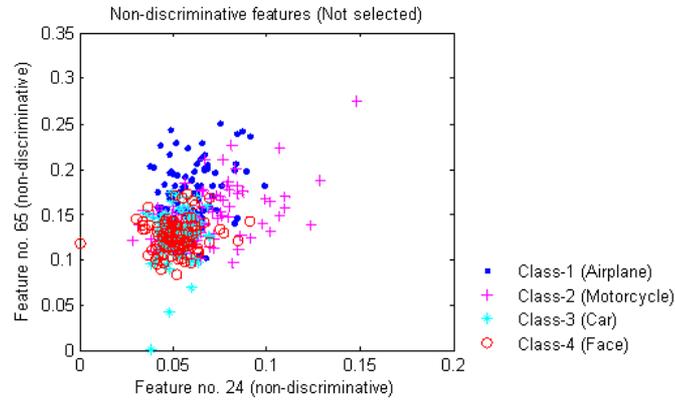
Figure A.12: Two dimensional distribution of samples in the object dataset, with x and y-axes representing two non-discriminative features. The features have high interclass overlap with the cluster centers closer to each other. Such features are discarded by the feature selection algorithm.

ilar to that shown in Fig. A.12 (which have higher interclass overlap) is difficult and may lead to misclassification. The proposed algorithm neglects such features, and excludes the corresponding rules from the classifier rule base. This increases the classification accuracy, and provides better margin of classification.

Once the feature values A_L and A_H are identified, the classification is done by the voting process using *Rule 1* and *Rule 2* (7 and 8).

[1] Z. Pawlak, Rough sets, International Journal of Computer and Information Science 11 (1982) 341–356.

[2] Z. Pawlak, Rough classification, International Journal of Man-Machine Studies 20 (1984) 469–483.

[3] L. A. Zadeh, Fuzzy sets, Information and Control 8 (3) (1965) 338–353.

[4] D. Dubois, H. Prade, Putting rough sets and fuzzy sets together, in: R. Slowinski (Ed.), Intelligent Decision Support: Handbook of Applications and Advances in Rough Sets Theory, Vol. 11 of Series D: System Theory, Knowledge Engineering and Problem Solving, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1992, pp. 203–232.

[5] D. Dubois, H. Prade, Rough fuzzy sets and fuzzy rough sets, International Journal of General Systems 17 (1990) 191–209.

[6] Z. Pawlak, Rough sets and fuzzy sets, in: Proceedings of ACM, Computer Science Conference, Nashville, Tennessee, 1995, pp. 262–264.

[7] D. Tian, J. Keane, X. Zeng, Evaluating the effect of rough sets feature selection on the performance of decision trees, in: Granular Computing, 2006 IEEE International Conference, 2006, pp. 57–62.

[8] A. A. Albrecht, Stochastic local search for the feature set problem,with applications to micro-array data, Applied Mathematics and Computation 183 (2006) 1148–1164.

[9] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, E. S. Lander, Molecular classification of cancer: Class discovery and class prediction by geneexpression monitoring, Science 286 (1999) 531–537.

[10] G. Piatetsky-Shapiro, P. Tamayo, Microarray data mining: Facing the challenges, SIGKDD Explorations 5 (2) (December, 2003) 1–5.

[11] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, T. Poggio, Robust object recognition with cortex-like mechanisms, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (3) (2007) 411–426.

[12] Y. C. Tsai, C. H. Cheng, J. R. Chang, Entropy-based fuzzy rough classification approach for extracting classification rules, Expert Systems with Applications 31 (2) (2006) 436–443.

[13] M. Sarkar, Fuzzy-rough nearest neighbor algorithms in classification, Fuzzy Sets and Systems 158 (2007) 2134–2152.

[14] A. Roy, K. P. Sankar, Fuzzy discretization of feature space for a rough set classifier, Pattern Recognition Letters 24 (2003) 895–902.

[15] X. Wang, J. Yang, X. Teng, N. Peng, Fuzzy-rough set based nearest neighbor clustering classification algorithm, Lecture Notes in Computer Science 3613/2005 (2005) 370–373.

[16] R. Jensen, C. Cornelis, A new approach to fuzzy-rough nearest neighbour classification, in: Proceedings of the 6th International conference on Rough sets and current trends in computing, 2008, pp. 310–319.

[17] S. Zhao, E. C. C. Tsang, D. Chen, X. Wang, Building a rule-based classifier-a fuzzy-rough set approach, IEEE Transactions on Knowledge and Data Engineering 22 (5) (2010) 624 – 638.

[18] M. Juneja, E. Walia, P. S. Sandhu, R. Mohana, Implementation and comparative analysis of rough set, artificial neural network (ann) and fuzzy-rough classifiers for satellite image classification, in: International

Conference on Intelligent Agent & Multi-Agent Systems, 2009. IAMA 2009., 2009, pp. 1–6.

[19] R. Jensen, Q. Shen, Fuzzy-rough data reduction with ant colony optimization, Fuzzy Sets and Systems 149 (1) (2005) 5–20.

[20] Q. Shen, A. Chouchoulas, A rough-fuzzy approach for generating classification rules, Pattern Recognition 35 (2002) 2425–2438.

[21] E. C. C. Tsang, S. Zhao, Decision table reduction in kdd: Fuzzy rough based approach, Transaction on Rough Sets, Lecture Notes in Computer Sciences 5946 (2010) 177–188.

[22] H. Qinghua, A. Shuang, Y. Daren, Soft fuzzy rough sets for robust feature evaluation and selection, Information Sciences 180 (22) (November, 2010) 4384–4400.

[23] F. Xu, D. Miao, L. Wei, Fuzzy-rough attribute reduction via mutual information with an application to cancer classification, Computers & Mathematics with Applications 57 (6) (March, 2009) 1010–1017.

[24] S. Chiu, Fuzzy model identification based on cluster estimation, Journal of Intelligent and Fuzzy Systems 2 (3) (September, 1994) 18–28.

[25] T. R. Golub, D. K. Slonim, C. G. M. Tamayo, P. Huard, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, E. S. Lander, Cancer program data sets (1999).
URL `http://www.broad.mit.edu/cgi-bin/cancer/datasets.cgi`

[26] G. J. Gordon, R. V. Jensen, L. Hsiao, S. R. Gullans, J. E. Blumenstock, S. Ramaswamy, W. G. Richards, D. J. Sugarbaker, R. Bueno, Supplemental information of gordon et al. paper (2002).
URL `http://www.chestsurg.org/publications/2002-microarray.aspx`

[27] J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, P. Meltzer, Microarray project (2001).
URL `http://research.nhgri.nih.gov/microarray/Supplement`

[28] J. Triesch, C. Malsburg, Robust classification of hand postures against complex backgrounds, in: Proceedings of the Second International Conference on Automatic Face and Gesture Recognition, 1996, Killington, VT, USA, October, 1996, pp. 170–175.

[29] A. S. Georghiades, P. N. Belhumeur, D. J. Kriegman, From few to many: Illumination cone models for face recognition under variable lighting and pose, IEEE Trans. Pattern Anal. Mach. Intelligence 23 (6) (2001) 643–660.

[30] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 2, 2003, pp. 264–271.

[31] C. C. Chang, C. J. Lin, Libsvm: a library for support vector machines (2001).
URL `http://www.csie.ntu.edu.tw/ cjlin/libsvm`

[32] T. Jirapech-Umpai, S. Aitken, Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes, BMC Bioinformatics 6:148.

[33] J. Lu, G. Getz, E. Miska, E. Alvarez-Saavedra, J. Lamb, D. Peck, A. Sweet-Cordero, B. L. Ebert, R. H. Mak, A. A. Ferrando, J. R. Downing, T. Jacks, H. R. Horvitz, T. R. Golub, Micro rna expression profiles classify human cancers, Nature 435 (June, 2005) 834–838.

[34] G. J. Gordon, R. V. Jensen, L. Hsiao, S. R. Gullans, J. E. Blumenstock, S. Ramaswamy, W. G. Richards, D. J. Sugarbaker, R. Bueno, Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma, Cancer Research 62 (September, 2002) 4963–4967.

[35] J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, P. Meltzer, Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks, Nature Medicine 7 (6) (June, 2001) 673–679.

[36] J. Xuan, Y. Wang, Y. Dong, Y. Feng, B. Wang, J. Khan, M. Bakay, Z. Wang, L. Pachman, S. Winokur, Y. Chen, R. Clarke, E. Hoffman, Gene selection for multiclass prediction by weighted fisher criterion, EURASIP Journal on Bioinformatics and Systems Biology 2007.

[37] S. L. Pomeroy, P. Tamayo, M. Gaasenbeek, L. M. Sturla, M. Angelo, M. E. McLaughlin, J. Y. H. Kim, L. C. Goumnerova, P. M. Black,

C. Lau, J. C. Allen, D. Zagzag, J. M. Olson, T. Curran, C. Wetmore, J. A. Biegel11, T. Poggio, S. Mukherjee, R. Rifkin, A. Califano, G. Stolovitzky, D. N. Louis, J. P. Mesirov, E. S. Lander, T. R. Golub, Prediction of central nervous system embryonal tumour outcome based on gene expression, Letters to Nature 415 (January, 2002) 436–442.

[38] T. Serre, L. Wolf, T. Poggio, Object recognition with features inspired by visual cortex, in: C. Schmid, S. Soatto, C. Tomasi (Eds.), Conference on Computer Vision and Pattern Recognition, San Diego, CA, 2005, pp. 994–1000.

[39] M. Turk, A. Pentland, Eigenfaces for recognition, Journal of Cognitive Neuroscience 3 (1991) 71–86.

[40] J. Triesch, C. Malsburg, Sebastien marcel hand posture and gesture datasets : Jochen triesch static hand posture database.
URL http://www.idiap.ch/resources/gestures/