A FILTER-BASED FEATURE SELECTION USING TWO CRITERION FUNCTIONS AND EVOLUTIONARY FUZZIFICATION

Jitwadee Chaiyakarn

A Dissertation Submitted in Partial
Fulfillment of the Requirements for the Degree of
Doctor of Philosophy (Computer Science)
School of Applied Statistic
National Institute of Development Administration
2013

A Filter-Based Feature Selection Using Two Criterion Functions and Evolutionary Fuzzification

Jitwadee Chaiyakarn

School of Applied Statistics

Assistant Professor.. Major Advisor

(Ohm Sornil, Ph.D.)

The Examining Committee Approved This Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy (Computer Science).

Associate Professor..Committee Chairperson

(Surapong Auwatanamongkol, Ph.D.)

Assistant Professor.. Committee

(Ohm Sornil, Ph.D.)

Associate Professor..Committee

(Pipat Hiranvanichakorn, Ph.D.)

Assistant Professor.. Committee

(Rawiwan Tenissara, Ph.D.)

Instructor .... Dean

(Siwiga Dusadenoad, Ph.D.)

June 2014

ABSTRACT

Title of Dissertation	A Filter-Based Feature Selection Using Two Criterion Functions and Evolutionary Fuzzification
Author	Miss Jitwadee Chaiyakarn
Degree	Doctor of Philosophy (Computer Science)
Year	2013

In information age, data has become increasingly large, in both dimension (the number of features) and volume. Data mining processes, such as data classification and data clustering, performed on high dimensional data can be time-consuming and can produce poor results due to the problem so called curse of dimensionality. Feature selection is one of the fundamental techniques that selects only the most significant features and eliminates irrelevant and redundant features from the entire set of features.

Filter-based feature selection is the technique to be focused in this dissertation. This technique can take less time to select significant features, especially for high dimensional data, but can not guarantee an optimal feature set.

Filter-based feature selection comprises of two important parts; searching process and criterion function evaluation. Floating search is commonly used for the searching process. It is a heuristic search, which does not take much time, however, can not guarantee an optimal feature set. The latter part relies on a criterion function, which is an independent measure to evaluate and select feature subsets without actually performing data mining algorithm. Therefore, it does not inherit any bias of the data mining algorithm. Usually, only one criterion function is used so one chararteristic of data is considered at a time. In this dissertation, two criterion functions are proposed for the feature evaluation. The two functions can compliment each other and two or more characteristics of data can be considered together to effectively select features.

Noise, ambiguity and uncertainty of data, which are frequently found in the real-world problem, can effect data mining process. Hence, fuzzy logic was applied to cope with these problems in this dissertation. A membership function was needed in the fuzzy logic to fuzzify original data and to infer data into fuzzy value. The fuzzy value was then passed through feature selection process instead of the original data. Genetic algorithm (GA) was used to determine the irregular shape of the membership function instead of by human expert.

From the experiments, the proposed two criterion functions was found to be effective to select features that can increase accuracy of data classification. The proposed method outperforms two existing methods, the hybrid and one criterion function filter-based methods. The experimental results also show that the proposed method with fuzzy logic enhances classification accuracy. It outperforms some wrapper-based feature selection methods, which have been widely known to achieve higher accuracy than filter-based methods.

The proposed feature selection method can also be used to reduce data dimension for unsupervised learning problems, such as data clustering. Unlike the supervised learning problems, there is no class label attribute of data objects to guide and cluster them into groups. Hence, it is not an easy task to select discriminant features for unsupervised learning problems. The criterion functions or measures for unsupervised learning problem w ere also proposed to be used for the proposed method. The experimental results showed that the proposed method can help improving clustering accuracy when compared with the results from other approaches. Therefore, the proposed feature selection method can be used for both supervised and unsupervised learning problems.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my greatest appreciation to my advisor, Assistant Professor Dr. Ohm Sornil. With his guidance, I have learned many different aspects in conducting research. I have learned how to find a good research topic and how to write a technical paper. Without his support and tremendous help, this dissertation would not be possible. I am also very thankful to my thesis committee, Associate Professor Dr. Surapong Auwatanamongkol, Associate Professor Dr. Pipat Hiranvanichakorn and Assistant Professor Dr. Rawiwan Tenissara. Their advice and suggestion have been very helpful.

I also would like to express my sincere gratitude to my family, who continuously supported and encouraged me until I could complete my doctoral degree.

Jitwadee Chaiyakarn
June 2014

TABLE OF CONTENTS

LIST OF TABLES

Tables **Page**

LIST OF FIGURES

Figures **Page**

SYMBOLS AND ABBREVIATIONS

Symbols

J_1	First criterion function
J_2	Second criterion function
S	Original feature set
S_{cand}	Candidate set
d	Predefined number of selected feature
D	Total number of original features
d_{sel}	Number of selected features in current set
d_{cand}	Number of features in a candidate set
S_{sel}	Selected feature subset
S_{cand}^{+}	Candidate set in the inclusion step
S_{cand}^{-}	Candidate set in the exclusion step

Abbreviations	**Equivalence**
AR	Average Redundancy
BAVE	Bhattacharyya Distance
CART	Classification and Regression Tree
CV	Cross Validation
CMI	Conditional Mutual Information
E	Entropy
GA	Genetic Algorithm

HCGA	Hierarchical Coevolutionary Genetic Algorithm
ISMF	Irregular-shape Membership Function
JMBH	Jeffery-Matusita Distance Bound to Bayes Error
KNN	K Nearest Neighbor
LS	Laplacian Score
LSE	Least Square Error
MAHA	Mahalanobis Distance
MF	Membership Function
MI	Mutual Information
NMI	Normalized Mutual Information
SFFS	Sequential Forward Floating Search
SFS	Sequential Forward Search

CHAPTER 1

INTRODUCTION

Advances in computer technology have led to the information age. Data collection becomes larger in both dimension (number of features) and volume. Some dimensions are irrelevant and some are redundant. Irrelevant dimensions can misguide machine learning results, especially when there are more irrelevant features than relevant ones. It may lead to low performance of data mining algorithm. An algorithm takes longer time to evaluate when the dimensions are redundant. In addition, noise, ambiguity and distraction could be mixed with data set; therefore, data mining technique is needed to cope with these problems. In this dissertation, we focus on the selection of predictive features or attributes to speed up t he learning process, improve the model generalization capability, and alleviate the effect of the curse of dimensionality.

1.1 Data Representation

Features in pattern recognition, in statistics, refer to individual measurable properties of the phenomena being observed. For example, in software design, it may be performance, portability or functionality, while in image processing, it may be color, texture or brightness. Properties must be something necessary or important to contribute to the discrimination of patterns or forms. The process of selecting the necessary properties is called feature selection. Its main objective is to retain the optimum 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. It is also known as variable selection, attribute selection or feature reduction.

The feature could be binary, nominal or numeric data. In most cases, features are represented by a vector composed of the values of some measurable features. Although all of these features constitute the inputs of a classifier, they have different impacts on the data mining algorithm performance.

1.2 Feature Selection

In many applications in computer vision, pattern recognition or data mining, one is often confronted with very high dimensional data. Large dimensionality presents a problem when handling the data due to the fact that, many commonly used operations are highly dependent on the level of dimensionality. Therefore, the time and spaces required for processing the data increase. Cai, Chang and He (2010: 333) stated that, “Various data mining and machine learning tasks, such as classification and clustering, that are analytically or computationally manageable in low dimensional spaces may become completely intractable in spaces of several hundred or thousand dimensions”. Most learning algorithms perform poorly in high dimensional space when there is small number of samples. This difficulty is known as the curse of dimensionality. Feature selection or dimensionality reduction plays a crucial rule to solve these problems. The low dimensional model is also easier for domain experts to interpret.

Feature selection algorithm seeks for a subset of features that can represent the characteristic of the data set. This subset will be used to generate data model. The main drawback of feature selection is the possibility of information loss. Useful information can be discarded if dimensionality reduction is done poorly. We can state that feature selection is an algorithm to select the most significant features and discard the least significant features to reduce evaluation time and sometimes improve effectiveness while minimizing the information loss.

1.2.1 General Feature Selection Procedure

A generalization of feature selection procedure is shown in Figure 1.1

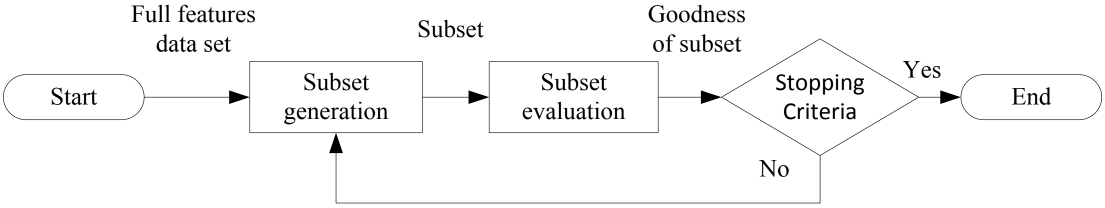

Figure 1.1 A Generalized Feature Selection Procedure

In Figure 1.1, or iginal data set will be passed through subset generation process to search for the next candidate feature subset to evaluate. Search strategy can be sequential search, random search or complete search. Sequential search strategy will add or remove one feature at a time until stopping criteria is met. This is a hill-climbing strategy to generate selected subset. Random search strategy randomly selects feature subset and then perform sequential search. Another way of this strategy is to totally select feature subset randomly to evaluate. For a complete search, all combinations of features should be covered. Even though it guarantees to find the optimal subset, the search is not exhaustively complete (Liu and Yu, 2005: 495).

Searching process could start with an empty set, which is called forward search, and then continually pick unselect feature to form a new subset. On the contrary, this process could start with full feature set and repeatedly remove irrelevant or redundant features to form a new subset. This approach is called backward search. Some researches combine both forward and backward strategies to generate a new subset, such as floating search.

The candidate feature subsets are evaluated according to certain evaluation criteria to get the best feature subset. These criteria can be categorized into two groups of independent and dependent. Independent criterion tries to measure the quality of feature subset. The quality measure is independent to data model. The best quality of feature subset will be selected. Information measure, distance measure and dependency measure are examples of the independent criterion. On the other hand, dependent criterion trains data model and evaluates feature subset through the model. Feature subset generating the best data model will be selected.

Stopping criteria could be any of the followings,

1) Selected subset with number of features equal to the predefined value
2) New subset of the feature does not yield a better result
3) Number of iteration is reached

The process of subset generation and evaluation is repeated until a given stopping criterion is satisfied.

1.2.2 Categories of Feature Selection Algorithm

According to subset evaluation, we could categorize feature selection algorithm into three groups of wrapper, filter and hybrid method.

Wrapper method is a kind of dependent criterion that incorporates the classification itself into the feature evaluation process. To evaluate the importance of a candidate feature subset, a classification model is built and used to evaluate the set. Filter method is a kind of independent criterion, which relies on general characteristics of the data to evaluate and select feature subsets without involving any data mining algorithm; therefore, it does not inherit any bias of the data mining algorithm. The hybrid method takes advantage of both wrapper and filter methods. It applies the filter-based technique to preselect highly significant features, and it applies a wrapper-based technique to add candidate features and evaluate the candidate sets in order to select the best one.

Some researches have divided feature selection of unlabeled data into two groups of Global and Local method. Guan, Dy and Jordan (2011: 2) stated that, "Global method selects a single set of features, whereas local method selects subsets of features, one subset for each cluster (where features in different clusters can vary)."

1.3 Categories of Learning Methods

Machine learning methods can be grouped into: supervised, unsupervised and semi-supervised. In supervised learning, class label of training data is provided as a guideline to generate model. The information helps verify if the prediction is correct or not. The supervised learning algorithm analyzes the training data and produces an inferred function, which can be used to map new examples. An optimal scenario will allow the algorithm to correctly determine the class labels for unseen instances;

whereas, in unsupervised learning, this information is missing. The target of this scenario is thus to discover the natural grouping structure of the data. In semi-supervised learning, only some of the data objects are labeled. It uses both labeled and unlabeled to train the model. Many researchers in the field of machine learning have found that unlabeled data, when used in conjunction with small amount of labeled data, can produce considerable improvement in learning accuracy.

1.4 Cross Validation

Typically, data used for model generation is divided into 2 groups, training data and test data. Training data is for model training and test data is for model evaluation. Data set does not provide independent test set separately; we have to split it into these two groups. The popular way that is often used for splitting is the k-fold cross validation (k-fold CV). The repeatability of results on new data can be assessed with this approach. The data samples are randomly divided into K non-overlapping subsets of the same size. One of the K subsets is "held-out" for testing, the model is trained on t he remaining K 1 subsets and an estimate of the accuracy can be obtained from model evaluation through its corresponding test set. This process repeats K times, so that each subset is treated once with the test set, and the average of the resulting K accuracy estimation forms the model accuracy.

1.5 Genetic Algorithm

Genetic algorithm (GA) was formally introduced in the 1970s by John Holland at the University of Michigan. GA is an adaptive heuristic search algorithm premised on the evolutionary ideas of natural selection and genetic. They were introduced as a computational analogy of adaptive systems. They are modeled loosely on the principles of the evolution via natural selection. A population of individuals or candidate solution to optimization problem is evolved toward better solution. A fitness function is used to evaluate individuals, and reproductive success varies with fitness. The fitness function correlates to the objective of optimization being solved.

GA model is a part of genetic evolution. The characteristics of individuals are expressed using genotypes, which can be mutated or altered. At first, population is randomly generated and each iteration is called a generation. In each generation, fitness value is evaluated in each individual. The fitter the individuals, the more chance to survive to the next generation; at the same time, others will be recombined or randomly mutated to generate a new generation, then the process iterates. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

1.5.1 Selection

A new population is selected at the end of each generation to serve as the population for the next generation. The new population can be selected only from the offspring, or from both the parents and the offspring. The selection operator should ensure that good individuals do survive to next generations.

1.5.2 Genetic Operator

Individuals are altered and mutated through application of crossover and/or mutation operator to generate offspring. Crossover operator concept is based on parents that are superior and have more opportunities to reproduce offspring with good material. Mutation operator concept is based on weak individual. It will result in introducing better traits to them, thereby increasing their chances of survival.

1.6 Overview

In this dissertation, we focus on filter-based feature selection algorithm, which can be used for both of supervised and unsupervised learning. The paper is organized into six chapters. In chapter 2, related works in the literatures are reviewed, while in chapter 3, we discuss about an improved filter-based feature selection for classification using two criterion functions. The algorithm applies two criterion functions with floating search strategy. In chapter 4, we introduce an evolutionary fuzzy feature selection, which applies fuzzy logic and GAs to select features to improve classification performance. In chapter 5, we represent that the proposed method can be applied to unsupervised clustering. The conclusions of this dissertation and recommendations for future work are addressed in chapter 6.

CHAPTER 2

LITERATURE REVIEW

2.1 Feature Selection for Classification

According to subset evaluation, we can categorize feature selection algorithm into three groups, which are wrapper, filter and hybrid methods.

2.1.1 Wrapper-Based Method

Wrapper method incorporates classification into feature evaluation process; therefore it is expected to return a subset of features that yields high performance results since every candidate feature set is evaluated by the classifier that will be used for the problem.

Sánchez-Maroño, Alonso-Betanzos and Castillo (2005: 515) propose a wrapper-based feature selection by using ANOVA decomposition and functional networks to evaluate global sensitivity index. Sobol ANOVA decomposition is used to generate functional component, then functional networks is used to select families of functions. The family with better performance results is selected for the basic function. Features with high index values are then selected from global sensitivity index, which is the result from the basic function.

Zhuo, Zheng, Wang, Li, Ai and Qian (2008: 397) present a GA based wrapper method for classification of hyper-spectral data using support vector machine (SVM). GA optimizes both the feature subset and SVM kernel parameters. Chromosome content consists of the feature subset and kernel parameter. They use a single objective function that combines the two criteria, classification accuracy and number of selected features into one to create fitness function for GA evaluation. Feature subset with high accuracy and small number of selected features is then selected.

Leng, Valli and Armstrong (2010: 167) use GA and K nearest neighbor (KNN) to rank importance of features, and top rank features are selected. Chromosome content that represents the subset of features is randomly selected. Classification accuracy using KNN is the fitness function to evaluate chromosomes. Chromosome is chosen for mutation by using roulette wheel selection strategy. Crossover had not been used in this study due to duplicate features in two chromosomes.

Since classification models are trained and tested many times, data becomes larger in dimensionality and in number of instances, this approach, therefore, takes too much time, and in many cases is inapplicable.

2.1.2 Filter-Based Method

Filter method relies on general characteristics of the data to evaluate and select feature subsets without involving any data mining algorithm thus it does not inherit any bias of the data mining algorithm. It composes of two important components: a selection algorithm and a criterion function. In selection part, all features are ranked and top features are selected to generate subset; at the same time, searching strategy can also be used to generate subset. Both of them use criterion function to evaluate feature or subset.

The filter-based method runs faster; consequently, it is more preferable for real-world problems, especially those with large data sets. Instead of performing classification as part of the feature selection process, quality measure is used to evaluate each feature set. The measure can be independent from the classification model, but it should be suitable for the problem. The filter approach takes less time than the wrapper approach since classifier is not trained and tested as many time as in the wrapper approach. Many researches found that it yields lower performance than the other two mentioned approaches; however, it is not true to state that the filter approach always gives lower accuracy. Some measures may provide equivalent or better performance than others to guide the search process.

Yu and Liu (2003) propose a fast filter forward selection method, which selects good features for classification by applying a new concept called predominant correlation without pairwise correlation analysis. A correlation measure is based on the information-theoretical concept of entropy to measure uncertainty of a random variable. It uses a correlation measure to select features relevant to classes, which are not redundant with other selected features.

Zhou, Weng, Wu and Schmidt (2003: 156) propose a forward algorithm to select features using conditional maximum entropy modeling. This study approximates entropy of unselected feature based on the model from the previous stage. The unselected feature, which maximizes the gain of the log likelihood is then selected. Because models from the previous stage are not all possible for the selection, it reduces evaluation time.

Fleuret (2004: 1531) use conditional mutual information (CMI) as the criterion function to speed up the forward search process. Algorithm will iteratively select one feature at a time, which maximizes its mutual information with class to predict, conditionally to the response of the features that have already been picked. Because this criterion function takes the feature that has already been selected into account, it could determine redundant feature and select only informative and discriminative features.

The first three review papers are forward search, which initially select an empty subset and then select one feature at a time. On the contrary, backward search starts with the entire set of features before discarding one feature at a time. Haindl, Somol, Ververidis and Kotropoulos (2006: 570) propose a backward filter-based feature selection method based on mutual correlation, similarity measure between two variables, to select uncorrelated features. The concept of this method is that, if two variables were independent, they should also be uncorrelated. An average absolute mutual correlation of a feature is evaluated over feature subset. The Feature with the largest value will be discarded at each iteration step until the remaining set is lower than or equal to a predefined number of the selected value.

2.1.3 Hybrid Method

The hybrid approach takes advantage of both wrapper and filter approach. It applies the filter-based technique to preselect highly significant features and applies the wrapper-based technique to add candidate features and evaluate candidate sets to select the best one.

Zhang, Wang, Zhao and Yang (2003: 380) apply a ReliefF algorithm to estimate the quality of attributes according to how well their values distinguish between the instances that are close to each other, and then use GA with classifier accuracy as its fitness function to search for an optimal feature subset. GA initialization is based on the descending order of the features according to the evaluation using ReliefF. Although ReliefF is fast and can be used with any data type, fairly noise-tolerant and unaffected by feature interaction, it does not handle well with redundant features.

Somol, Novovičová and Pudil (2006: 634) present a hybrid-floating search, hSFFS, by applying a filter criterion function to filter some features and generate a candidate set before applying a wrapper criterion function to select the best one from the candidate set. They also present the backward counterpart, hSBFS, which is a wrapper-dominating hybrid method. This hybrid scheme uses only a fraction of full wrapper computational time to obtain results. Its performance relies on hybridization coefficient. Although it takes longer time when higher coefficient is used, higher performance will be obtained.

Gan, Hasan and Tsui (2011: 281) propose an alternative to hSFFS, which is a filter-dominating hybrid method where a filter criterion is used to select the best feature from an unselected set, while a wrapper criterion is only used to evaluate a feature subset. They stated that it is difficult to choose appropriate value of hybridization coefficient to control the proportion of features in candidate sets, pre-selected by a filter, and passed to a wrapper the same way as in wrapper dominating. They propose the use of only one filter to select a feature to be added or to be removed and then use a wrapper to compare the selected best feature subsets. The experimental result shows that the proposed method gives comparable result with wrapper dominating while taking less time.

2.2 Sequential Forward Floating Search (SFFS)

Pudil, Novovičová and Kittler (1994: 1119) introduce sequential forward floating search (SFFS) using a criterion function to select a feature and compare candidate subsets. SFFS can be classified as a wrapper or a filter approach depending on a criterion function used. It performs well; nevertheless, its computation time is high and the data set is very large. The structure of SFFS is in Figure 2.1.

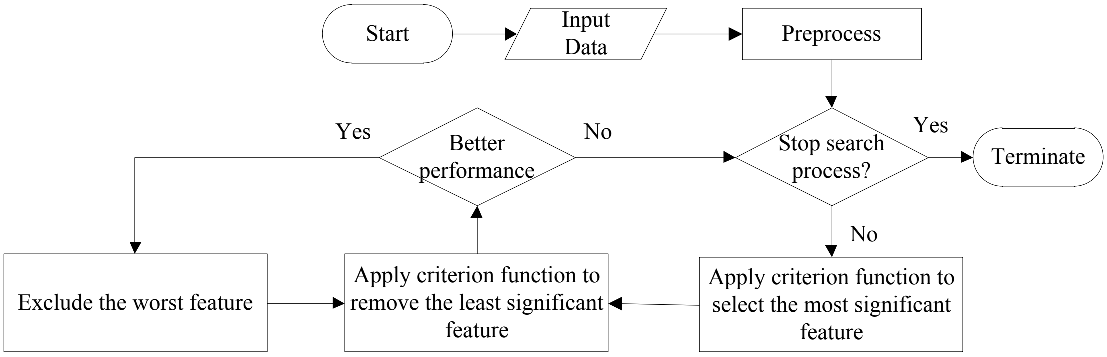
Start
Input Data
Preprocess
Stop search process?
Yes
Terminate
No
Apply criterion function to select the most significant feature
Apply criterion function to remove the least significant feature
Better performance
Yes
No
Exclude the worst feature

Figure 2.1 The Structure of SFFS Algorithm

In Figure 2.1, SFFS consists of two phases: forward search and backward search. The forward search selects the best unselected feature according to a criterion function to form new subset, and the backward search iteratively determines which members of the selected subset are to be removed if the remaining set improves the performance according to the same criterion function in forward search. The algorithm loops back to forward search until the stopping condition is reached. Pseudo-code of SFFS is shown in Figure 2.2.

```
Input: $S$ is a feature set, $d$ is a predefined number of selected features, $J$ is a criterion function, and $J_k$ is the criterion function value of $k$ selected features.
Output: $S_{sel}$ is the selected feature set
Initialize: $k = 0$ and $S_{sel} = \phi$
(1) Inclusion
    a. If $k = d$ then terminate
    b. $x^+ = \arg\max_{x \in S \backslash S_{sel}} J(S_{sel} \cup x)$
    c. $S_{sel} = S_{sel} \cup x^+, k = k + 1$
(2) Conditional exclusion
    a. $x^- = \arg\max_{x \in S_{sel}} J(S_{sel} \setminus x)$
    b. If ($J(S_{sel} \setminus x^-) > J_{k-1}$) then
        i. $S_{sel} = S_{sel} \setminus x^-, k = k - 1$
        ii. Go to (2)
    Else go to (1)
```

Figure 2.2 Pseudo-code of SFFS Algorithm

In Figure 2.2, the algorithm starts with forward search with an empty selected subset. A number of selected features are determined as stopping condition. Each unselected feature is temporarily added to current subset and then evaluates their criterion value. The best feature, which makes the new subset better than the previous one, will be selected. In conditional exclusion step, criterion value of the features in the new subset is evaluated by temporarily excluding it from selected subset. The feature, which makes the remaining set better, will be removed.

Improved versions of SFFS have been proposed in various researches to provide better performance.

Somol, Pudil, Novovičová and Pacliík (1999: 1157) present ASFFS, which keeps selecting features to add into and remove from the list. ASFFS represents a more sophisticated version of classical floating search algorithm. Instead of single feature adding or removing, user has a possibility to let the algorithm perform a more thorough search with better chances to find the optimal solution by setting higher generalization level. ASFFS adds one feature at a time until the number of features reaches a certain point. To remove a feature, a reverse of the adding is applied.

Songyot Nakariyakul and Casasent (2009: 1933) present IFFS that adds an additional step to check whether replacing a weak feature can improve the criterion function value. The extra step is performed after the removing step. This step conditionally removes one feature at a time and use sequential forward search (SFS) to select an unselected feature and add it to each resultant feature set. If the replacement helps improve the performance of the feature subset, the algorithm will step back to find another feature to remove again until the replacement can no longer improve any criterion function and then goes back to the inclusion to select a new feature from unselected ones.

Sun, Wang, Zhang and Zhao (2010: 2862) propose an improvement to SFFS without the need to predefine the number of selected features. Mutual information and Parzen window estimator are the criterion functions to handle a mixture of continuous and categorical input features and continuous target features. Instead of using appropriate number of selected features, they define two non-negative thresholds to guarantee that the variation of information in each forward or backward step is significant enough. The algorithm terminates automatically when there is not enough variation in mutual information obtained from adding or removing any feature.

SFFS has a limitation on high dimensional (e.g., hundreds) feature selection especially for wrapper-based (Gan, Hasan and Tsui, 2011: 280). Although using wrapper-dominating hybrid method could improve efficiency and save time, it is too computational expensive for high dimensional data set and it may lead to impractical feature. In the next chapter, we propose filter-based feature selection with SFFS to accelerate the algorithm. We use two criterion functions to guide the search process and also to improve efficiency.

CHAPTER 3

IMPROVING FILTER-BASED FEATURE SELECTION FOR CLASSIFICATION USING TWO CRITERION FUNCTIONS

A filter-based method normally takes less time than do bot h hybrid and wrapper methods while most of the time yields less classification accuracy. We attempts to improve the quality of the feature set selected from a filter-based method in order to achieve high classification accuracy. We improve upon the filter-based SFFS by employing two criterion functions with different characteristics to complement each other and allowing more efficient searches for features by introducing candidate sets.

3.1 The proposed Method

Structure of the proposed technique is shown in Figure 3.1. The algorithm begins with the inclusion step that employs the first function (J_1) to create a s et of candidates for inclusion and employs the second function (J_2) to select a subset of features where a feature selected is one when combined with those previously selected features of size k gives the best value of J_2, forming a s elected subset of size $k+1$. Comparing this newly formed subset with the previously best subset of size $k+1$, the algorithm retains the better one.

Next, in the exclusion step J_1 is applied to rank features upon the benefits of their removals and create targets for exclusion. The J_2 function then determines the feature to be removed from the set to give the best feature set of size k. If the new subset is better than the previously selected set, the exclusion step retains the better one and iterates to smaller subsets, or else the algorithm goes back to the inclusion step. Calculation by function J_2 usually takes a long time. J_1 which runs faster helps screening candidate features thus reduces the amount of calculations to be performed by J_2.

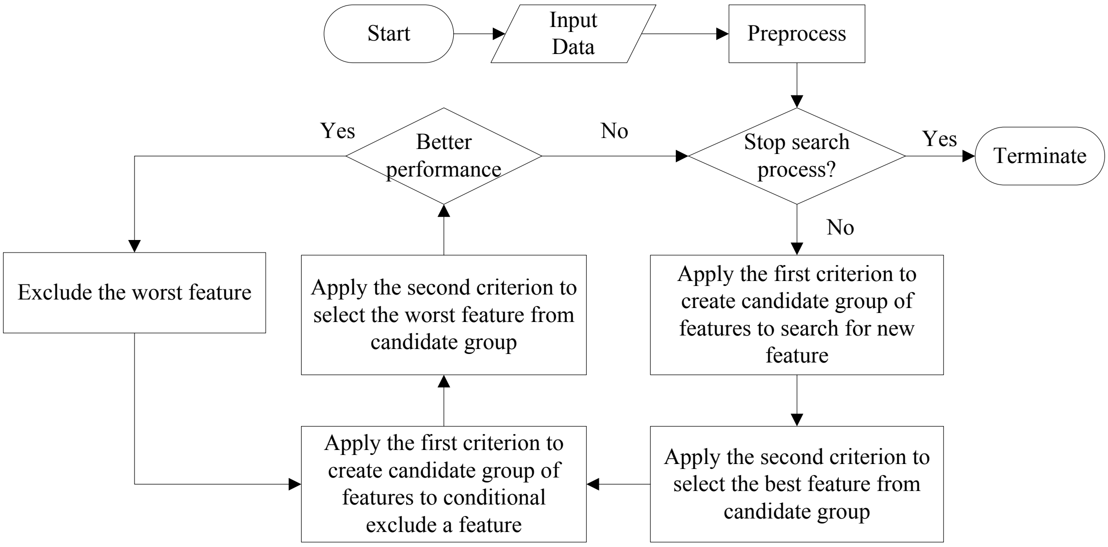
Start
Input Data
Preprocess
Stop search process?
Yes
Terminate
No
Apply the first criterion to create candidate group of features to search for new feature
Apply the second criterion to select the best feature from candidate group
Apply the first criterion to create candidate group of features to conditional exclude a feature
Apply the second criterion to select the worst feature from candidate group
Better performance
Yes
No
Exclude the worst feature

Figure 3.1 The Proposed Feature Selection Method.

Conditional mutual information (CMI) (Fleuret, 2004: 1539) is employed as the first criterion function. It is a measure of dependency between two variables with respect to a class, conditional to the response of features already picked. CMI selects features which maximize mutual information to target class where such information must not have been caught by features already picked to reduce redundant features. It is used to generate a candidate set of features which are suitable to be added to or removed from the selected subset. Using the candidate sets allows more efficient searches over the features. The J_2 function then selects a feature to be added to or removed from these sets, instead of considering every unselected feature every time.

In this section, the algorithm will be described using the following notations. S is the original feature set, S_{cand} is a candi date set, d is the predefined number of selected features, J_1 is the first criterion function, and J_2 is the second criterion function. D is the total number of original features. d_{sel} is the number of selected features in current set. d_{cand} is the number of features in a candidate set. S_{sel} is the selected feature subset. S_{cand}^{+} is the candidate set in the inclusion step, and S_{cand}^{-} is the candidate set in the exclusion step.

CMI can be computed as follows:

$$I(Y;X_n \mid X_m) = H(Y,X_m) - H(X_m) - H(Y,X_n,X_m) + H(X_n,X_m) \qquad (3.1)$$

where H is an entropy function, Y is a class attribute, X_m is a set of features, and X_n is the feature to be selected. The J_1 function in the inclusion step is computed as follows:

$$J_{11}(X_n) = I(Y;X_n \mid S_{sel}) \text{ where } X_n \in S \setminus S_{sel} \qquad (3.2)$$

In the inclusion step, unselected features are evaluated using J_{11} , select d_{cand} unselected features that yields minimum value of J_{11} and sort them in a descending order according to their values of J_{11}. A candidate set is generated as follows:

$$S_{cand}^{+} = \{x_i \mid x_i \in S \setminus S_{sel} \text{ and } i = [1..d_{cand}] \text{ and } J_{11}(x_1) \geq J_{11}(x_2) \geq \ldots \geq J_{11}(x_{d_{cand}})\} \qquad (3.3)$$

A feature to be removed must be the one providing the least information to the target classes, and its information has been caught by features already picked. Therefore, J_1 in the exclusion step is computed as follows:

$$J_{12}(X_n) = I(Y;X_n \mid S_{sel} \setminus X_n) \text{ where } X_n \in S_{sel} \qquad (3.4)$$

In the exclusion step, the selected features are evaluated using J_{12} , select d_{cand} selected features that yields maximum value of J_{12} and sort and sorted in an ascending order according to the values of J_{12}. A candidate set is generated as follows:

$$S_{cand}^{-} = \{x_i | x_i \in S_{sel} \text{ and } i = [1..d_{cand}] \text{ and } J_{12}(x_1) \leq J_{12}(x_2) \leq ... \leq J_{12}(x_{d_{cand}})\} \qquad (3.5)$$

The algorithm for the proposed method is described in Figure 3.2. It begins with an empty selected subset (S_{sel}). In the inclusion step, the candidate set (S_{cand}^{+}) is constructed by equation (3.3). The feature selected is the one when combined with the previously selected subset of size *k* gives the best subset when evaluated with *J₂*, forming the selected subset of size *k+1*. Then the algorithm compares the new subset with the previously selected subset of size *k+1* and retains the better one.

In the exclusion step, the candidate set (S_{cand}^{-}) is created by equation (3.5). The feature to be removed is the one when removed from the selected subset yields the best subset with *k* features according to J_2. The algorithm compares the new subset and the previously selected subset of size *k* and retains the better one. The exclusion step continues to smaller subsets if the new subset is better, or else the algorithm goes back to the inclusion step. The algorithm terminates when the selected subset size is $d_{sel} + \Delta$. In the pseudo-code, for every feature in S_{sel}, we store the best values of J_1 and J_2 and their corresponding feature subsets in a lookup table to speed up t he calculation.

```
Let $S_d$ be subset containing $d$ features
Initialization
        $S_{sel}$ = empty set
        $d = 0$
Begin
        Step 1: Inclusion step
                Use (3.3) to generate $S_{cand}^{+}$
                $x_i = \arg\max_{x \in S_{cand}^{+}} J_2(S_{sel} \cup x)$
                If $J_2(S_{sel} \cup x_i) > J_2(S_{d+1})$ then
                        $S_{sel} = S_{sel} \cup x_i$
                        $S_{d+1} = S_{sel}$
                        $J_2(S_{d+1}) = J_2(S_{sel} \cup x_i)$
                Else
                        $S_{sel} = S_{d+1}$
                $d = d + 1$
        Step 2: Exclusion step
                Use (3.5) to generate $S_{cand}^{-}$
                $x_i = \arg\max_{x \in S_{cand}^{-}} J_2(S_{sel} \setminus x)$
                If $J_2(S_{sel} \setminus x_i) > J_2(S_d)$ then
                        $S_{sel} = S_{sel} \setminus x_i$
                        $S_{d-1} = S_{sel}$
                        $J_2(S_{d-1}) = J_2(S_{sel})$
                        $d = d - 1$
                        Go to step 2
                Else
                        Go to step 3
        Step 3: Stopping criterion checking
                If $d = d_{sel} + \Delta$ then exit an algorithm
                Else go to step 1.
End.
```

Figure 3.2 The Proposed Algorithm

3.2 The Second Criterion Function

Since the aim of the proposed technique is for the classification task, 4 measures are studied as the J_2 criterion function which consists of:

3.2.1 Mutual Information (MI)

MI is the information measure. It measures mutual dependence of two random variables. It is defined as relative entropy which measures uncertainty of random variable; it is also viewed as a measure of impurity in data (Fleuret, 2004: 1539). MI can be calculated as follows:

$$I(Y;X_n) = H(Y) + H(X_n) - H(Y,X_n)$$

where H is entropy function, Y is a class attribute, and X_n is the feature to be selected. The effect of MI is close to CMI used in J_2; it is used in this research to provide a baseline performance.

3.2.2 Bhattacharyya Distance (BAVE)

BAVE measures similarity between two probability distributions which is suitable for measuring distance between classes (Bruzzone and Serpico, 2000: 552). BAVE can be calculated as follows:

$$B_{ave} = \sum_{i=1}^{c}\sum_{j=1}^{c} P(\omega_i)P(\omega_j)B_{ij}$$

$$B_{ij} = \frac{1}{8}(m_i - m_j)^t \left(\frac{\sum_i + \sum_j}{2}\right)^{-1}(m_i - m_j) + \frac{1}{2}\log\left[\frac{\left(\frac{\sum_i + \sum_j}{2}\right)}{\sqrt{|\sum_i||\sum_j|}}\right]$$

where m_i, m_j and $\sum_i, \sum_j$ are mean vectors and covariance matrices for the class ω_i and ω_j, respectively.

3.2.3 Jeffreys-Matusita Distance Bound to the Bayes Error (JMBH)

J-M distance is a measure of statistical separability for two-class cases and can be extended to multi-classes (Bruzzone, Roli and Serpico, 1995: 1319). It measures how much the two probability distribution functions are separated (Pereira et al., 2007: 251). JMBH is similar to J-M distance but based on the Bhattacharyya upper bound to the Bayes error probability (Bruzzone and Serpico, 2000: 553). JMBH can be calculated as follows:

$$J_{bh} = \sum_{i=1}^{c}\sum_{j=1}^{c}\sqrt{P(\omega_i)P(\omega_j)}J_{ij}^2$$

$$J_{ij} = \left[2(1-e^{-B_{ij}})\right]^{1/2}$$

3.2.4 Mahalanobis Distance (MAHA)

MAHA measures similarity between an unknown sample set and a known one, based on correlation between variables (De Maesschalck, Jouan-Rimbaud and Massart, 2000: 14). It takes into account the global distribution and a rough approximation of the intra-class distance through the difference between the means. MAHA can be calculated as follows:

$$D_M(x) = \sqrt{(x-\mu)^T S^{-1}(x-\mu)}$$

where μ is the mean vector and S is the covariance matrix for a group.

Mahalanobis distance is a particular case of the Bhattacharyya distance when the standard deviations of the two classes are the same.

A good measure will make the classes far apart, thus BAVE and MAHA are chosen because of its ability to select a feature subset that can maximize inter-class distances. JMBH considers not only interclass distances but also similarity to error probability behavior (Bruzzone and Serpico, 2000: 553). MI is a special case of relative entropy to select a feature set that gives high purity for each class. These measures will be studied in the experiments.

To calculate CMI which categorical data is expected, numeric features are discretized by a modification of the method presented in (Tsai, Lee and Tang, 2008) where instead of using all distinct values, we consider the values that appear more than once as cut points. This change makes discretization faster, and experiments show that the modification gives higher performance than does the method presented in (Tsai, Lee and Tang, 2008).

3.3 The Classifier

Classification and Regression Trees (CART) was introduced by Breiman, Friedman, Olshen and Stone (1984). CART is based on the fundamental idea that each split should be selected so that the data in each descendant subset is purer than the data in the parent node. The node impurity is largest when all classes are equally mixed together and smallest when the node contains only one class. CART produces binary splits. Hence, it produces a binary tree. CART uses Gini impurity index as an attribute selection measure to build a decision tree. Consider a parent node *t*, which contains the data that belongs to the j^{th} class. The impurity function it for node t is given by:

$$i(t) = 1 - \sum_j p^2(j \mid t)$$

The decrease of impurity of split is given by:

$$\Delta i(t) = i(t) - p_L i(m_L) - p_R i(m_R)$$

where *t* is a parent node which is split into two nodes m_L and m_R. CART will search through all possible values of all variables for the best split which maximizes the decrease of impurity $\Delta i(t)$.

We evaluate the efficiency of our method with CART. We compare accuracy of classifier based on the features it selects to accuracy with the same classifiers build on features selected by other techniques.

3.4 Experimental Results

To select features, we apply CMI as the first criterion function and four measures are studied as the second criterion function. Data with only selected features and class attribute is passed through CART to obtain classification accuracy. For a data set that does not provide a separate test set, a 10-fold cross validation is applied.

3.4.1 Data Sets

Data used in the experiments are 20 standard data sets with various sizes from the UCI machine learning repository (Asuncion and Newman, 2007). Five data sets consist only of categorical attributes, and fifteen data sets include numeric attributes. Detail of all data sets is shown in Table 3.1.

Table 3.1 Data Sets Used in Experiment

Name	Feature Type	A number of instances	A number of features
Wine	Numeric	178	13
Image Segmentation	Numeric	2310	19
Breast Cancer	Numeric	569	30
Ionosphere	Numeric	351	34
Dermatology	Numeric	366	34
Soybean	Categorical	307	35
Lung Cancer	Categorical	32	56
Promoter	Categorical	106	57
Spambase	Numeric	4601	57
Sonar	Numeric	208	60
Splice	Categorical	3190	60
Libras Movement	Numeric	360	90
Hill valley with noise	Numeric	1212	100
Hill valley without noise	Numeric	1212	100
Musk	Numeric	6598	166
Musk2	Numeric	6598	166
Semeion	Categorical	1593	256
Madelon	Numeric	2600	500
Isolet	Numeric	7797	617
Multivariate	Numeric	2000	649

3.4.2 Effects of the Candidate Set Size

We introduce the candidate set to form a group of some unselected features that is good enough to be selected. These features will be evaluated with J_2 to select the best one. The candidate set makes algorithm faster because instead of evaluating all unselected feature with J_2 , only features in candidate set are evaluated. Size of the candidate set d_{cand} defines the search space for features to be included and excluded in the inclusion and the exclusion steps, respectively. We vary the values of d_{cand} with different data sets and find that the value of d_{cand} greater than 5 doe s not give significantly different performance. The accuracy of every configuration with d_{cand} = 1 and d_{cand} = 5 is shown in Table 3.2.

We can see that using d_{cand} = 5 yields higher accuracy than does using d_{cand} = 1 in 72 ou t of 80 e xperiments (90%), across different configurations and data sets. This is because not only focusing on high quality features to be selected, candidate set also makes algorithm more thorough search to get high predictive features. Therefore, in further experiments all configurations are assumed to use d_{cand} = 5.

Table 3.2 Effects of Different Candidate Set Sizes

Data Set	**CMI+MI**		**CMI+JMBH**		**CMI+BAVE**		**CMI+MAHA**	
	d_{cand}=1	d_{cand}=5	d_{cand}=1	d_{cand}=5	d_{cand}=1	d_{cand}=5	d_{cand}=1	d_{cand}=5
Wine	88.89	88.89	100	100	88.89	94.44	88.89	88.89
Image Segmentation	90.29	90.62	90.05	90.19	90.05	90.14	90.62	90.62
Breast Cancer	94.74	92.98	94.4	96.49	94.4	96.49	94.4	96.49
Ionosphere	97.14	97.14	94.29	100	94.29	100	94.29	97.14
Dermatology	100	100	97.22	97.22	97.22	97.22	97.22	97.22
Soybean	89.53	90.54	90.2	90.2	90.2	90.2	90.2	90.2
Lung Cancer	100	100	100	100	100	100	100	100
Promoter	90.91	90.91	90.91	90.91	90.91	90.91	90.91	90.91
Spambase	93.49	93.49	93.71	93.71	93.71	93.28	93.71	93.71
Sonar	80.95	80.95	80.95	85.71	80.95	85.71	80.95	80.95
Splice	93.1	93.1	93.42	93.28	93.42	93.28	93.42	93.42
Libras Movement	77.78	77.78	66.67	75	66.67	77.78	69.44	80.56
Hill valley with noise	56.6	56.6	56.6	56.6	56.6	56.6	56.6	56.6
Hill valley without noise	57.26	57.1	57.76	57.76	56.93	57.59	57.43	57.43
Musk	97.88	98.33	97.58	98.33	97.58	98.33	97.58	97.27
Musk2	97.12	96.82	96.97	97.27	96.97	97.27	96.97	96.82
Semeion	79.38	79.38	81.25	79.38	81.25	81.25	83.13	81.88
Medelon	68.5	69.83	68.5	74	68.5	74	70.33	76
Isolet	71.46	73.57	71.46	71.46	71.46	71.46	71.01	71.65
Multivariate	97.5	97	97	98	97	97	97	97

3.4.3 Effectiveness of the Second Criterion Function (J_2)

The task of the second criterion function (J_2) is to maximize inter-class distances and at the same time to minimize intra-class distances. In this section, a set of experiments is performed to evaluate the effectiveness of using two objective functions (the proposed method) against the use of a single function (an original filter-based method) and to study the effectiveness of different J_2 criterion functions. The results are shown in Table 3.3. For each data set, classification accuracy is shown in the upper row, and the number of selected features is shown in the lower row. We can see that feature selection is beneficial for classification; in 15 out of 20 data sets there is at least one configuration with feature selection that gives higher accuracy and less number of features than does the original data set.

Comparing between using one and two criterion functions, for each corresponding function (except only MI) we can see that using two functions in the way proposed in this research gives higher accuracy than using one function. In addition, different J_2 functions yield different results for both of selected features and classification accuracy. CMI+MI gives the highest performance in 5 data sets, CMI+JMBH gives the highest performance in 9 da ta sets, CMI+BAVE gives the highest performance in 7 data sets, and CMI+MAHA gives the highest performance in 8 data sets. We can see that CMI+JMBH yields the best overall results.

Together, the use of CMI and JMBH improves classification accuracy. Because JMBH not only maximizing inter-class distance, it also limits error probability bound to be minimum. In addition, JMBH directly relate to Bhattacharyya distance exhibits a s aturating behavior for large distance values while BAVE continues to increase significantly even when the topological distance between them reaches values corresponding to well-separated classes (Bruzzone and Serpico, 2000: 553). For MI, it considers only the information to target class not including class distance or even error bound. MAHA only takes into account the global distribution and a rough approximation of the intra-class distance through the difference between the means. Thus, in the further experiments we will use CMI+JMBH to represent the proposed method.

Table 3.3 Performance of Different Feature Selection Configurations (The Highest Accuracy for Each Data Set is bold font)

Data set		Original Data	Filter-Based Approach				Proposed Approach			
			MI	**JMBH**	**BAVE**	**MAHA**	**CMI+MI**	**CMI +JMBH**	**CMI +BAVE**	**CMI +MAHA**
Wine	Accuracy	83.33	88.89	**100**	94.44	94.44	88.89	**100**	94.44	88.89
	Features	14	2	2	4	11	2	2	4	2
Image Segmentation	Accuracy	90.29	**90.71**	85.24	85.24	90.33	90.62	90.19	90.14	90.62
	Features	20	12	5	5	10	8	9	9	5
Breast Cancer	Accuracy	89.47	94.74	91.23	91.23	**96.49**	92.98	**96.49**	**96.49**	**96.49**
	Features	31	19	2	2	10	7	6	6	10
Ionosphere	Accuracy	88.57	94.29	94.29	94.29	97.14	97.14	**100**	**100**	97.14
	Features	35	3	3	3	9	5	9	9	17
Dermatology	Accuracy	94.44	**100**	72.22	77.78	97.22	**100**	97.22	97.22	97.22
	Features	35	19	11	10	22	19	8	14	14
Soybean	Accuracy	90.54	**90.54**	29.05	29.05	36.82	**90.54**	90.2	90.2	90.2
	Features	36	15	1	1	6	15	17	17	17
Lung Cancer	Accuracy	66.67	**100**	66.67	33.33	33.33	**100**	**100**	**100**	**100**
	Features	57	1	2	1	2	2	4	4	4
Promoter	Accuracy	81.82	**90.91**	72.72	72.72	81.82	**90.91**	**90.91**	**90.91**	**90.91**
	Features	58	13	2	2	2	8	6	6	6
Spambase	Accuracy	92.41	**93.93**	88.29	88.07	92.84	93.49	93.71	93.28	93.71
	Features	58	38	18	17	51	17	27	9	27
Sonar	Accuracy	71.43	80.95	71.43	71.43	**85.71**	80.95	**85.71**	**85.71**	80.95
	Features	61	26	6	6	7	6	18	18	6

Table 3.3 (Continued)

Data set		Original Data	Filter-Based Approach				Proposed Approach			
			MI	JMBH	BAVE	MAHA	CMI+MI	CMI +JMBH	CMI +BAVE	CMI +MAHA
Splice	Accuracy	91.22	93.1	70.21	70.21	62.38	93.1	93.28	93.28	**93.42**
	Features	61	17	2	2	1	17	7	7	7
Libras Movement	Accuracy	69.44	75	52.78	52.78	63.89	77.78	75	77.78	**80.56**
	Features	91	36	4	4	25	31	9	9	22
Hill valley with noise	Accuracy	55.45	56.44	49.67	49.67	55.94	**56.6**	**56.6**	**56.6**	**56.6**
	Features	101	4	3	3	11	5	5	5	5
Hill valley without noise	Accuracy	**60.07**	57.59	52.48	52.48	59.08	57.1	57.76	57.59	57.43
	Features	101	46	2	2	44	25	16	28	22
Musk	Accuracy	96.82	97.42	95.15	95.15	95.75	98.33	**98.33**	**98.33**	97.27
	Features	167	42	9	9	10	37	18	18	43
Musk2	Accuracy	**98.03**	96.06	96.06	96.06	95	96.82	97.27	97.27	96.82
	Features	167	40	13	13	22	39	12	12	18
Semeion	Accuracy	**81.88**	72.5	61.25	65	78.75	79.38	79.38	81.25	**81.88**
	Features	257	47	18	19	45	36	47	47	47
Medelon	Accuracy	75.67	70.5	54.83	54.83	72.17	69.83	74	74	**76**
	Features	501	48	23	23	6	30	18	18	19
Isolet	Accuracy	**79.6**	63.76	63.75	63.75	61.57	73.57	71.46	71.46	71.65
	Features	618	45	44	44	43	45	48	48	41
Multivariate	Accuracy	95	97	88	83.5	92.5	97	**98**	97	97
	Features	650	24	5	14	23	17	13	25	11

3.4.4 Comparison with Other Approaches

In this section, the proposed technique is compared against the hybrid and the filter approaches that use JMBH as the second criterion function. The resulting running time is shown in Figure 3.3 and Figure 3.4 while the accuracy values are shown in Table 3.4. It can be seen that the running time for every technique is proportional to the size of the data set.

Due to hybrid method requires training and testing of models during the second phase of the method, the proposed method runs faster than a hybrid method for all data set (and increasingly faster with larger data sets). Comparing with original filter method, the proposed method runs faster for large data set size since a number of evaluations of the second criterion function in original method which is equal to a number of unselected features while it is equal to a number of candidate features in the proposed method. This is shown in Figure 3.3.

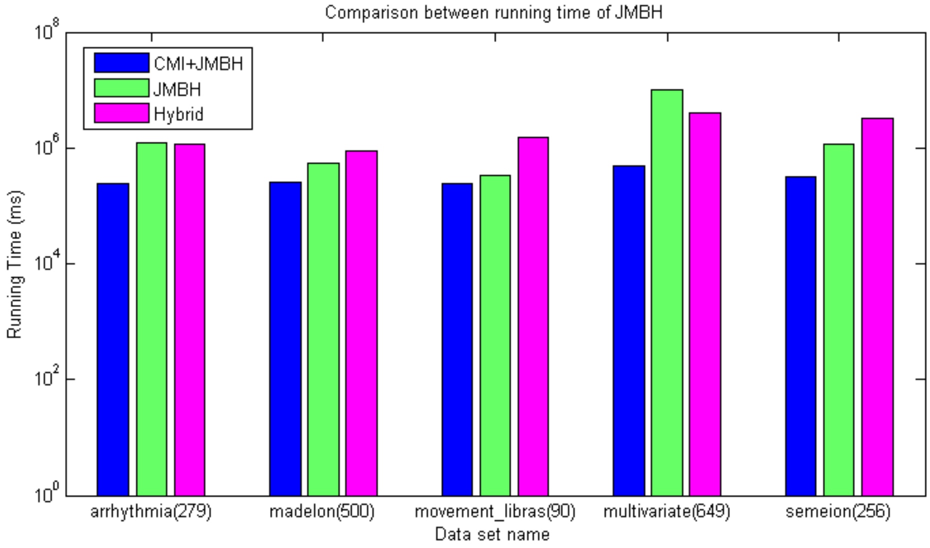

Figure 3.3 Running Time of Three Feature Selection Approaches for Large Data Set

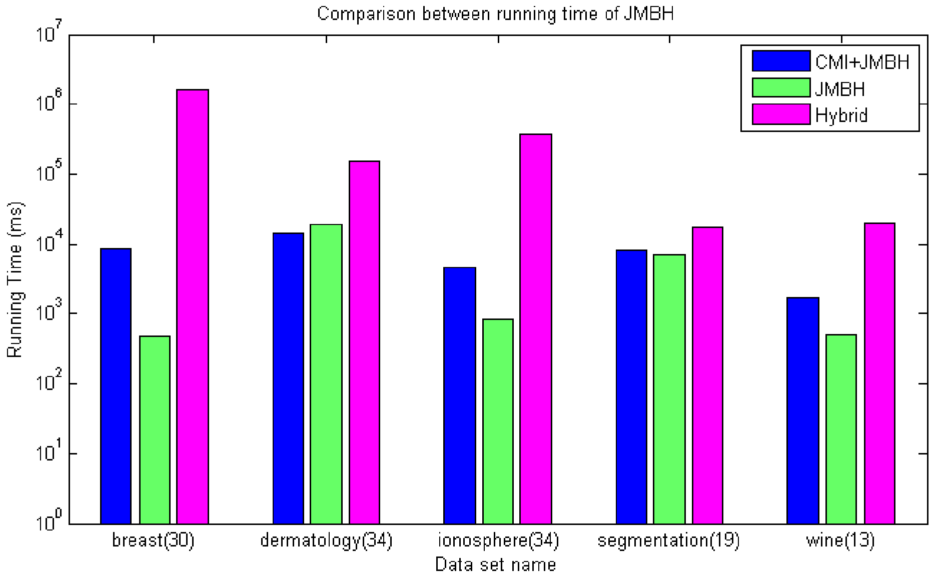

Figure 3.4 Running Time of Three Feature Selection Approaches For Small and Moderate data set

In terms of effectiveness (in Table 3.4), the proposed method outperforms the original filter-based method in 19 out of 20 data sets. The hybrid method performs the best in 10 data sets while the proposed method performs the best in 11 data sets plus 3 data sets, each with the accuracy equal to that of the best method but with a larger set of selected features.

Table 3.4 Comparisons of the Three Feature Selection Approaches (The Highest Accuracy for Each Data Set is bold font)

Data set		Hybrid method	Filter Method	Proposed method
Wine	Accuracy	**100**	**100**	**100**
	Features	2	2	2
Image Segmentation	Accuracy	**92.57**	85.24	90.19
	Features	3	5	9
Breast Cancer	Accuracy	**96.49**	91.23	**96.49**
	Features	4	2	6
Ionosphere	Accuracy	97.14	94.29	**100**
	Features	17	3	9
Dermatology	Accuracy	**100**	72.22	97.22
	Features	6	11	8
Soybean	Accuracy	**91.55**	29.05	90.2
	Features	10	1	17
Lung Cancer	Accuracy	**100**	66.67	**100**
	Features	2	2	4
Promoter	Accuracy	**90.91**	72.72	**90.91**
	Features	3	2	6
Spambase	Accuracy	92.41	88.29	**93.71**
	Features	17	18	27
Sonar	Accuracy	80.95	71.43	**85.71**
	Features	9	6	18
Splice	Accuracy	**94.98**	70.21	93.28
	Features	9	2	7
Libras Movement	Accuracy	72.22	52.78	**75**
	Features	9	4	9
Hill valley with noise	Accuracy	64.19	49.67	**56.6**
	Features	13	3	5
Hill valley without noise	Accuracy	**64.03**	52.48	57.76
	Features	25	2	16
Musk	Accuracy	97.73	95.15	**98.33**
	Features	16	9	18
Musk2	Accuracy	96.82	96.06	**97.27**
	Features	16	13	12
Semeion	Accuracy	78.62	61.25	**79.38**
	Features	39	18	47
Medelon	Accuracy	**83.5**	54.83	74
	Features	24	23	18
Isolet	Accuracy	56.51	63.75	**71.46**
	Features	44	44	48
Multivariate	Accuracy	97	88	**98**
	Features	35	5	13

3.5 Discussion

Determining value of d_{cand} is important. If we define it too high, algorithm takes much time but does not yield higher accuracy. If we define it too low, search space is not wide enough. The selected features will dominate only on the first criterion function not complement between two criterion functions and we do not obtain high accuracy. From experiment results, using $d_{cand} = 5$ yields higher accuracy than does using $d_{cand} = 1$, across different configurations and data sets. It is because not only focusing on high quality features to be selected; candidate set also makes algorithm more efficient search to select high predictive features.

From the experiment results, different criterion functions yields different results because each function has a unique characteristic and we can see that JMBH as the second criterion function yields the best result. Therefore if we properly select the second criterion function, we can get the best results. The proposed method provides more options for users to select two suitable criterion functions according with solving problem. This provides more opportunity to get high accuracy. The results show that the proposed technique outperforms the original filter which uses only single criterion function. The proposed technique also outperforms the hybrid approach with the same reason. Although wrapper-based part in hybrid approach can improve accuracy but it is only used to evaluate feature subset not to be guidance for searching high predictive features. Therefore the proposed technique improves accuracy and reduces computation time in searching process.

In this chapter we evaluate data set by using original data in feature selection process. It provides low accuracy in some data sets. We suppose that it can be caused by mixtures of noisy data or uncertainty data. Therefore in the next chapter we preprocess these data sets to reduce the effect of such data to improve accuracy.

CHAPTER 4

FILTER-BASED FUZZY FEATURE SELECTION USING GENETIC ALGORITHM

In real-world applications, ambiguous and noisy data are very common. These imperfect data can lead to an inaccurate model from machine learning process. Fuzzy Logic, which is a multi-value logic that allows intermediate values to be defined between conventional crisp evaluations, e.g., true/false, yes/no, etc., provides a simple way to define conclusions based upon vague, ambiguous, imprecise, noisy, or missing input information (Engelbrecht, 2007: 143). Hence, fuzzy logic can be used to handle the imperfect data while minimize the losses of information due to its processes. (Grande, Suárez and Villar, 2007: 57).

In this chapter we propose a feature selection technique for classification, that uses fuzzified feature values instead of the original values. The fuzzification on each feature value is performed using irregular-shaped membership functions evolved by a genetic algorithm.

4.1 Literature Review

From the benefits of fuzzy logic in handling ambiguous and noisy data, fuzzy set theory has been used to improve performance of many feature selection algorithms.

For filter-based feature selection (Grande, Suárez and Villar, 2007: 57) use fuzzy mutual information as the objective function to select features for classification problems with predefined fuzzy partitions for all of the features. Features with maximum information about the desired output are selected. They use the extended Battiti feature selection algorithm for regression problem.

Li and Wu (2008: 218) use fuzzy extension matrix as a searching strategy to map crisp data into fuzzy space and select features. The extension matrix is used with membership degree of fuzzy feature instead of original data. They use a triangular shape MF to fuzzify original data. Features that minimize inter-class similarity with respect to selected fuzzy subset and maximize goodness-of-fit are selected.

Jalali, Nasiri and Minaei (2009: 718) apply fuzzy feature selection with a greedy search and a consistency measure to select features. The proposed method projects original data into fuzzy space by using a triangular or sigmoid MF. Forward feature selection works with fuzzy features and uses a consistency measure as the criterion function. The best feature in each iteration is the feature that maximizes the consistency measure.

For wrapper-based feature selection (Cintra, Martin, Monard and Camargo, 2009: 214) propose a wrapper feature selection method with wrapper-based fuzzy rule generation. GA employs triangular shape MF to fuzzify original data. This work applies the backward search strategy with fuzzy features to generate the fuzzy rule bases (FRBs). FRBs error rate is the objective function for feature selection.

Hedjazi, Kempowsky-Hamon, Despènes, Le Lann, Elgue and Aguilar-Martin (2010: 6827) apply fuzzy feature selection and a classification technique to select an optimal feature (called sensor) subset. Centered binomial MF is chosen in this work. Membership Margin Based Attribute Selection (MEMBAS) is the feature selection algorithm used in this work. Classification error is the objective function for feature selection.

Vieira, Sousa and Kaymak (2012: 6) propose a feature selection method using an ant colony optimization algorithm to solve multi-objectives optimization problems in feature selection. They adapt the classical criteria in model based feature selection, by describing them using membership functions. In this work a Gaussian shape MF is used as fuzzy criteria for classification error minimization and trapezoidal shape MF is used as fuzzy criteria for feature cardinality minimization. The evaluation of subset solution is obtained by using an aggregation operator to combine the two fuzzy criteria which are the number of features minimization and classification error rate minimization into a single objective function.

4.2 The Proposed Method

In fuzzy logic, an element belongs to a set or class to a degree, indicating the certainty (or uncertainty) of membership. A membership function (MF), also referred to as the characteristic function of the fuzzy set, is used in the fuzzification process. It receives crisp data and produces the membership degrees (Engelbrecht, 2007: 470). An inference process maps the level of membership received from the fuzzification process and produces a fuzzified output for each fuzzy value.

4.2.1 A Filter-Based Irregular-Shaped Membership Function Generation

MF for fuzzy sets can be of any shape or type as determined by experts in the domain over which the sets are defined. In a fuzzy system, there are several patterns of MF such as triangular, trapezoidal and Gaussian-shaped whose determination requires skill and knowledge about information to select and specify appropriate pattern to use.

The proposed method employs fuzzification of original feature values where feature values are fuzzified using irregular-shaped membership functions (ISMFs) evolved by a genetic algorithm. The fuzzified feature values are then used in the feature selection process proposed in chapter 3.

Determination of an MF shape (for data fuzzification) by experts may not be possible for a big data set containing a large number of features. We adapt the wrapper-based hierarchical co-evolutionary genetic algorithm proposed by (Huang, Pasquier and Quek, 2007) to generate ISMFs where a criterion function is used as the fitness function for the GA. ISMF is a highly-generic type of MF, which is composed of unevenly spaced sampling points, connected together with straight line segments. Hierarchical Co-evolutionary Genetic Algorithm (HCGA) is adopted to determine the number of sampling points and their values.

An ISMF shape is represented as one pivot point, left shoulder points and right shoulder points depicted in Figure 4.1. The ISMF of each fuzzy set for each input variable will be represented in genetic segmentations and concatenated into one chromosome in the first level for the corresponding variable. The pivot point is the point where the membership value is the maximum. It is encoded as offsets (distance) along x and y axes from the pivot point of the preceding membership function of the same feature. Except for the first membership function, the pivot point is encoded as offsets from the starting value of the feature. The left points and right points are all encoded as the offsets from the respective pivot point. When evolving with GA, coordinates of all the point values will be changed and so the shape of MF. A chromosome in the second level composes of genes pointing to chromosomes for all variables in L1-level. Content in L2-level gene is an integer value of index of L1-level chromosome. The structure of two-level HCGA chromosome populations is depicted in Figure 4.2.

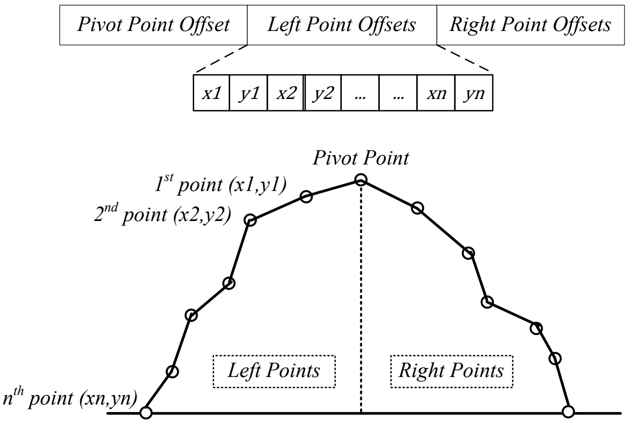

Figure 4.1 A Genetic Segment Representing An ISMF Shape

The algorithm partitions and encodes possible solutions as populations in different levels, allowing for different kinds of chromosomes and genetic operators. A higher level chromosome selects a set of lower-level chromosomes to form a solution. In this case, a highly complicated search task can be properly partitioned into several subtasks, which are simultaneously and effectively handled.

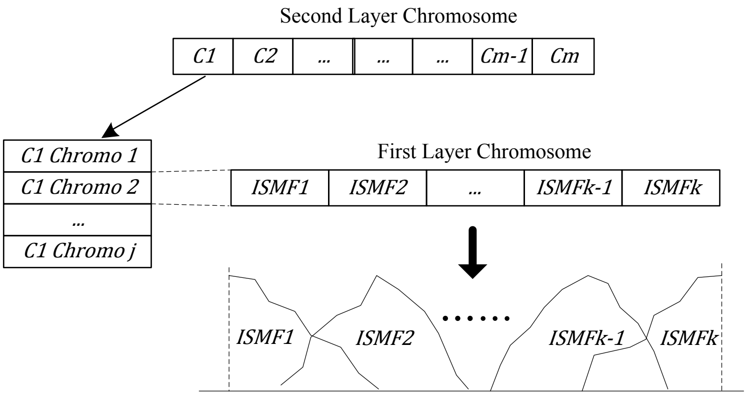

Figure 4.2 Structure of Two-Level HCGA Chromosome Populations for Deriving ISMFs

An example ISMFs is shown in Figure 4.3. T here are eight fuzzy sets of Alcohol feature. The leftmost set has no left shoulder point and the rightmost set has no right point. Each set is represented by an integer value that the leftmost set is 1 and the rightmost set is 8.

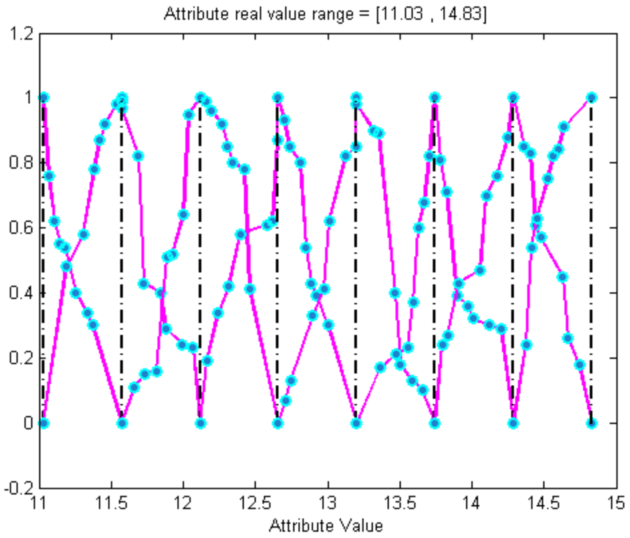

Figure 4.3 An Example of ISMFs for Alcohol Feature in the Wine Dataset

Genetic operations, which are selection, crossover and mutation, in the two levels can be different. In this dissertation, selection methods and crossover operators are not predefined for both levels. Combinations of selection methods and crossover operators are explored. The selection methods include Roulette, Tournament, Uniform and Stochastic uniform selection. The crossover operators include Single-point, Two-point, Arithmetic and Scatter crossover operator. Uniform mutation operator is applied for both levels because uniform mutation operator can be used for integer value genes in the second level and floating value genes in the first level. All genes are mutated with equal probability.

A fitness function is used to evaluate how close a given design solution (referred to as a chromosome) is to achieve the fuzzy set objectives. In this dissertation, we apply the second criterion function used in filter-based feature selection process as the fitness function for the GA. For example, if we apply mutual information (MI) as the second criterion function in feature selection, we will use MI as the fitness function of HCGA. Since the objective of the second criterion function is to evaluate a feature subset and select the best one, it is appropriate to be used for evaluating the fitness of a chromosome for deriving ISMFs.

An example to infer the fuzzy set value of a variable is depicted in Figure 4.4. Let suppose we want to get the fuzzy set value of a variable value of 12.99. We draw the vertical line from the point of 12.99 on the x-axis to cross the variable ISMFs. It can be concluded that the given value belongs to the 5th fuzzy set since its membership degree is higher than is the 4th fuzzy set. Therefore, the fuzzy value of 12.99 is 5.

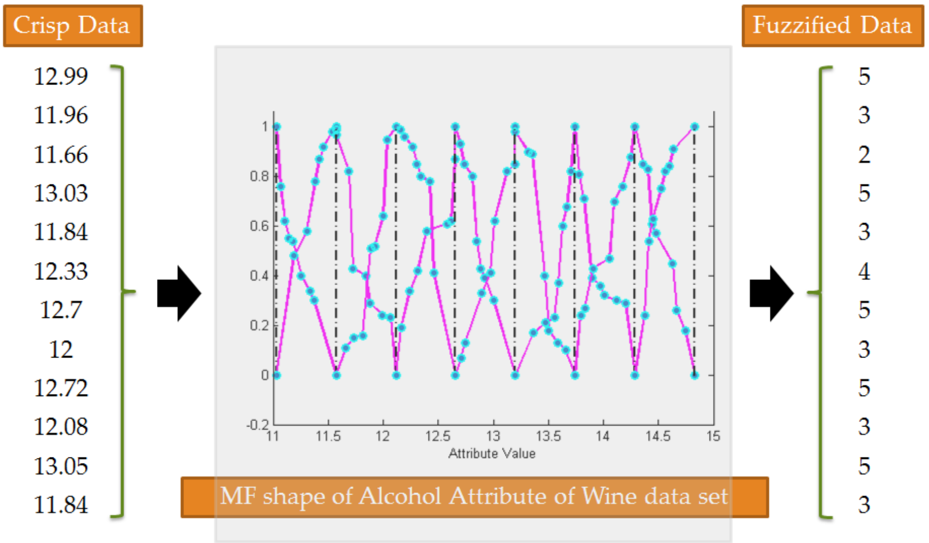

Figure 4.4 An Example of Inferring Fuzzified Feature Values from Given Crisp Feature Values Using Given ISMFs.

During crossover and mutation operations of the GA, constraints and repairing schemes are employed to ensure the genetically derived ISMFs of fuzzy sets are proper and valid.

1) Range repair: to ensure that all ISMF points are covered by ISMFs.

2) Nearby ISMFs repair: to ensure that there is no significant overlap between two ISMFs.

3) Membership value repair: to ensure that all points have non-negative membership degrees.

4) Completeness repair: to ensure that the last point on either side has a zero membership degree.

5) Out-bound cross point repair: to ensure that every two nearby ISMFs overlap at a proper membership degree.

Regarding out-bound cross point repairing, from experiments we found that if two consecutive shapes have pivot point values close to the new horizontal offset value obtained using the formula provided in the paper (Huang, Pasquier and Quek, 2007); the repaired point can be moved to the next point as shown in Figure 4.5. The result is a membership value of the point between the repaired point and the next point will have more than one value. Thus, in this research the midpoint is used instead.

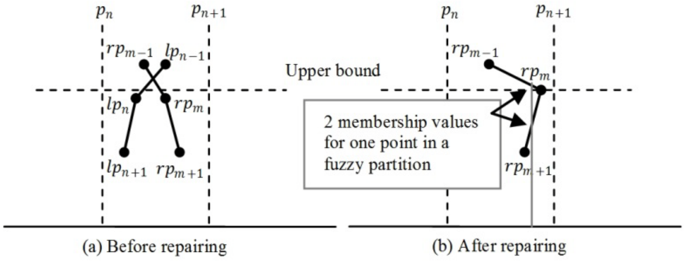

Figure 4.5 Out-Bound Cross Point Repairing. There Are Two Membership Values of Crisp Data Between rp_m and rp_{m+1}

4.2.2 Feature Selection Algorithm

We use fuzzified output of features in the feature selection process instead of original data which could be a mixture of ambiguous and noisy data (as used in chapter 3). The feature selection process with fuzzy logic is shown in Figure 4.6.

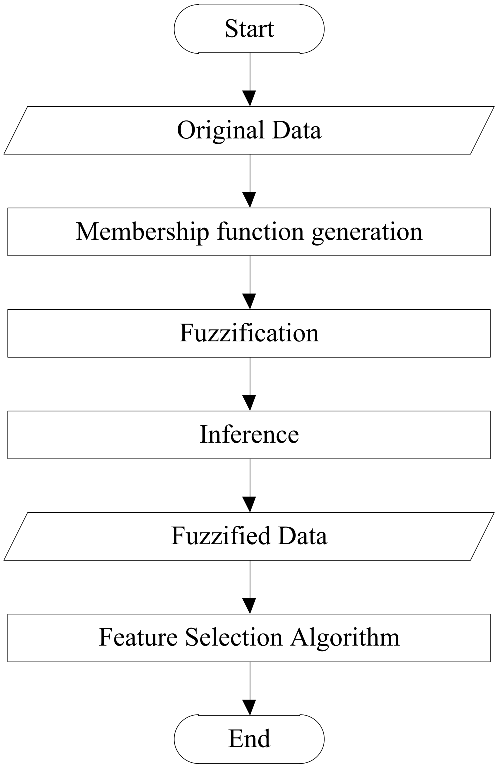

Figure 4.6 Feature Selection Process

4.3 The Proposed Algorithm

Structure of the proposed technique is shown in Figure 4.7. We modified the algorithm presented in previous chapter to incorporate fuzzy logic. Therefore, before going into the selection step, irregular-shaped MFs are generated first, and then the fuzzified output is used to select features. We use two criterion functions where CMI is the first function, and three measures are studied as the second function.

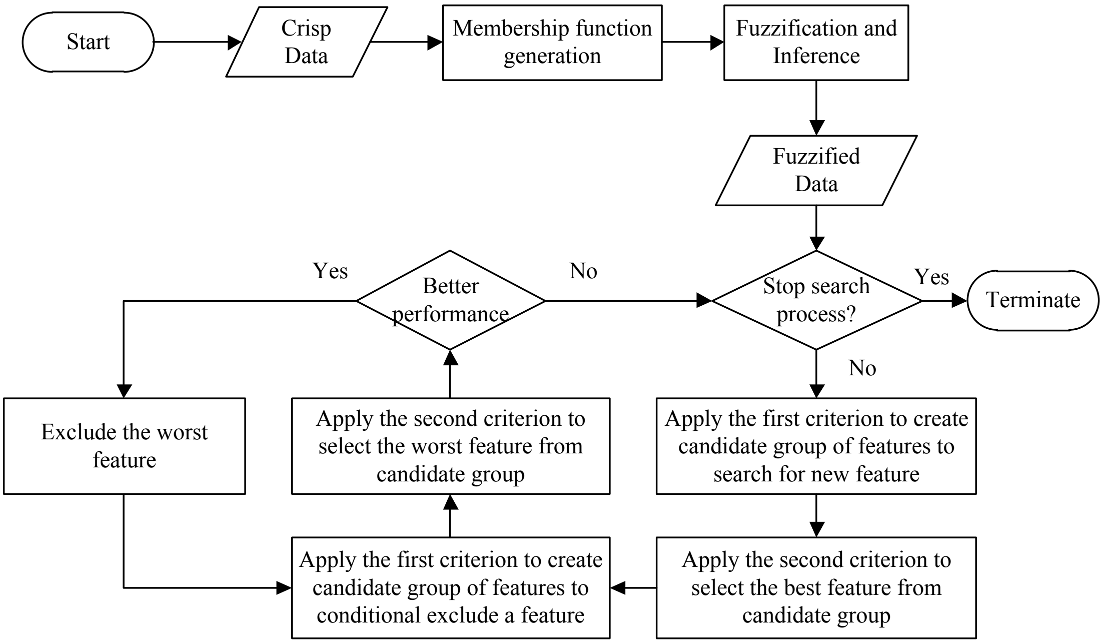
Start
Crisp Data
Membership function generation
Fuzzification and Inference
Fuzzified Data
Yes
Better performance
No
Stop search process?
Yes
Terminate
No
Exclude the worst feature
Apply the second criterion to select the worst feature from candidate group
Apply the first criterion to create candidate group of features to search for new feature
Apply the first criterion to create candidate group of features to conditional exclude a feature
Apply the second criterion to select the best feature from candidate group

Figure 4.7 Structure of Proposed Algorithm

The fuzzified output (fuzzy partition value) is a discrete value, and it is used to evaluate CMI. Therefore we do not discretize original data in this proposed algorithm as does in chapter 3.

4.4 The Second Criterion Function

We study three measures from the previous chapter as the second criterion function.

4.4.1 Mutual Information (MI)

MI can be calculated as follows:

$$I(Y;X_n) = H(Y) + H(X_n) - H(Y,X_n)$$

where H is entropy function, Y is a class attribute, and X_n is the feature to be selected. The effect of MI is close to CMI used in J_2; it is used in this research to provide a baseline performance.

4.4.2 Jeffreys-Matusita Distance Bound to the Bayes Error (JMBH)

JMBH can be calculated as follows:

$$J_{bh} = \sum_{i=1}^{c}\sum_{j=1}^{c}\sqrt{P(\omega_i)P(\omega_j)}J_{ij}^2$$

$$J_{ij} = \left[2(1-e^{-B_{ij}})\right]^{1/2}$$

$$B_{ij} = \frac{1}{8}(m_i - m_j)^t\left(\frac{\sum_i + \sum_j}{2}\right)^{-1}(m_i - m_j) + \frac{1}{2}\log\left[\frac{\left(\frac{\sum_i + \sum_j}{2}\right)}{\sqrt{|\sum_i||\sum_j|}}\right]$$

4.4.3 Mahalanobis Distance (MAHA)

MAHA can be calculated as follows:

$$D_M(x) = \sqrt{(x-\mu)^T S^{-1}(x-\mu)}$$

where μ is the mean vector, and S is the covariance matrix for a group.

4.5 Experimental Results

To select features, we apply CMI as the first criterion function, and three measures are studied as the second criterion function. Fuzzified data with only selected features and class attribute is passed through CART to obtain classification accuracy. For a data set that does not provide a separate test set, a 10-fold cross validation is applied.

4.5.1 Data Set

Data used in the experiments are 15 standard data sets with various sizes from the UCI machine learning repository (Asuncion and Newman, 2007). Details of the data sets are shown in Table 4.1.

Table 4.1 Data Sets Used in Experiments

Name	Feature Type	A number of instances	A number of features
Pima	Numeric	768	8
Wine	Numeric	178	13
Image Segmentation	Numeric	2310	19
Breast Cancer	Numeric	569	30
Ionosphere	Numeric	351	34
Dermatology	Numeric	366	34
Spambase	Numeric	4601	57
Sonar	Numeric	208	60
Libras Movement	Numeric	360	90
Hill valley with noise	Numeric	1212	100
Hill valley without noise	Numeric	1212	100
Musk	Numeric	6598	166
Musk2	Numeric	6598	166
Arrhythmia	Numeric	452	279
Madelon	Numeric	2600	500

4.5.2 Experiment Setup

After exploring various combinations of system parameters, the followings are used in all experiments. For the hierarchical co-evolutionary genetic algorithm, the population sizes of the first layer and the second layer are set as 100 and 30, respectively. The upper and lower bounds are set to ensure that two nearby ISMFs overlap at a proper membership degree. The values of the two bounds are set to 0.8 and 0.1, r espectively. We choose crossover operator from four operators (Single-point, Two-point, Arithmetic and Scatter crossover operator) for each layers of HCGA separately. Crossover rate of both layers is varied between 0.7 and 0.9. Uniform mutation operator is used for both layers with different mutation rates. Mutation rate for the first layer is varied between 0.1 and 0.3 while mutation rate for the second layer is varied between 0.01 and 0.03. The candidate set size for the feature selection d_{cand} is 5. The maximum number of ISMFs is 9 (for initialization).

4.5.3 Effectiveness of the Proposed Technique

This section, we study the benefit of feature selection using fuzzified feature values in comparison to the one using original feature values, the effectiveness of the proposed method, and the comparative performance of different 3 functions as the J_2 criterion. Three feature selection approaches, i.e., the original filter-based; non-fuzzified two-criterion (CMI+J_2 without fuzzy); and the proposed fuzzified two-criterion (CMI+J_2 with fuzzy), are studied. The results (in Table 4.2) show that feature selection when using fuzzified feature values in general gives higher accuracy than when using the original set of features; that the proposed feature selection algorithm is effective where in most cases two criterion CMI+J_2 without fuzzy yields higher accuracy than does using J_2 alone, except when using MI as J_2. Also, in almost all cases the fuzzified CMI+J_2 gives equivalent or better results than does the best of J_2 and CMI+J_2 without fuzzy. Regarding the J_2 functions in the fuzzified CMI+J_2 configuration, JMBH is found to perform equivalently or better than the best of all other functions in terms of classification accuracy in 11 out of 15 data sets, among those there are 2 data sets with accuracy equal to the best values but with smaller feature sets which include Dermatology (16 features for fuzzified CMI+JMBH and 22 features for fuzzified CMI+MI and fuzzified CMI+MAHA) and Spambase (40 features for fuzzified CMI+JMBH, and 52 f eatures for fuzzified CMI+MI). For the Breast Cancer data set, the fuzzified CMI+JMBH yields 98.24% with 4 features while the highest accuracy is 98.25% with 5 features achieved by the fuzzified CMI+MAHA. Further experiments will thus use JMBH as the J_2 criterion function.

Table 4.2 The Results of the Original Filter Method (J_2), CMI+J_2 Without Fuzzification and CMI+J_2 With Fuzzification

Data set		Original Data	MI			JMBH			MAHA		
			J_2	CMI+J_2 Without Fuzzy	CMI+J_2 With Fuzzy	J_2	CMI+J_2 Without Fuzzy	CMI+J_2 With Fuzzy	J_2	CMI+J_2 Without Fuzzy	CMI+J_2 With Fuzzy
Pima	Accuracy	70.13	75.33	75.33	75.33	74.03	76.63	**79.22**	75.33	75.33	75.33
	Features	8	3	3	3	5	3	4	3	3	3
Wine	Accuracy	83.33	88.89	88.89	94.44	**100**	**100**	**100**	94.44	88.89	**100**
	Features	13	2	2	3	2	2	3	11	2	2
Image Segmentation	Accuracy	90.29	90.71	90.62	**92.57**	85.24	90.19	**92.57**	90.33	90.62	**92.57**
	Features	19	12	8	5	5	9	9	10	5	5
Breast Cancer	Accuracy	89.47	94.74	92.98	96.49	91.23	96.49	98.24	96.49	96.49	**98.25**
	Features	30	19	7	2	2	6	4	10	10	5
Ionosphere	Accuracy	88.57	94.29	97.14	**100**	94.29	**100**	97.14	97.14	97.14	97.14
	Features	35	3	5	12	3	9	11	9	17	4
Dermatology	Accuracy	94.44	**100**	**100**	**100**	72.22	97.22	**100**	97.22	97.22	**100**
	Features	35	19	19	22	11	8	16	22	14	22
Spambase	Accuracy	81.82	**93.93**	93.49	93.48	88.29	93.71	93.48	92.84	93.71	93.26
	Features	58	38	17	52	18	27	40	51	27	41

Table 4.2 (Continued)

Data set		Original Data	MI			JMBH			MAHA		
			J_2	**CMI+** J_2 **Without Fuzzy**	**CMI+** J_2 **With Fuzzy**	J_2	**CMI+** J_2 **Without Fuzzy**	**CMI+** J_2 **With Fuzzy**	J_2	**CMI+** J_2 **Without Fuzzy**	**CMI+** J_2 **With Fuzzy**
Sonar	Accuracy	71.43	80.95	80.95	**95.23**	71.43	85.71	**95.23**	85.71	80.95	**95.23**
	Features	60	26	6	4	6	18	5	7	6	5
Libras Movement	Accuracy	69.44	75	75	75	52.78	75	80.56	63.89	80.56	**83.33**
	Features	91	36	29	21	4	9	11	25	22	36
Hill valley with noise	Accuracy	55.45	57.59	56.6	59.40	49.67	56.6	**59.90**	55.94	56.6	59.08
	Features	100	46	5	7	3	5	12	11	5	4
Hill valley without noise	Accuracy	60.07	57.59	57.1	**60.89**	52.48	57.76	**60.89**	59.08	57.43	60.23
	Features	101	46	25	12	2	16	14	44	22	10
Musk	Accuracy	96.82	97.42	**98.33**	98.03	95.15	**98.33**	**98.33**	95.75	97.27	97.58
	Features	167	42	37	48	9	18	42	10	43	39
Musk2	Accuracy	98.03	96.06	97.12	96.97	96.06	97.27	**97.58**	95	96.82	96.52
	Features	167	40	19	31	13	12	47	22	36	28
Arrhythmia	Accuracy	66.67	64.44	62.22	80	80	60	**82.22**	77.78	62.22	66.67
	Features	279	13	5	5	3	17	20	30	23	5
Medelon	Accuracy	75.67	70.5	69.83	78.33	54.83	74	**84.83**	72.17	76	81.5
	Features	500	48	30	7	23	18	12	6	19	9

4.5.4 Comparisons with Other Approaches

In this section, we compare the proposed method with other 3 feature selection approaches, namely, wrapper, filter, and hybrid approaches, using JMBH as the criterion function. The results (in Table 4.3) show that the proposed method (Fuzzified CMI+JMBH) gives the highest accuracy in 12 out of 15 data sets. Considering these 3 alternative approaches, we find that the wrapper approach performs the best, the hybrid approach performs moderately, and the filter approach gives the lowest accuracy. This confirms our earlier discussions regarding the relative performance of the feature selection approaches.

Table 4.3 The Results of The Wrapper Method, The Hybrid Method, The Original Filter Method, and The Proposed Method

Data set	Wrapper Method	Hybrid Method	Filter Method	Proposed method
Pima	75.33	75.33	74.03	**79.22**
	3	3	5	4
Wine	**100**	**100**	**100**	**100**
	2	2	2	3
Image Segmentation	**92.57**	**92.57**	85.24	**92.57**
	3	3	5	9
Breast Cancer	96.49	96.49	91.23	**98.24**
	3	4	2	4
Ionosphere	**97.14**	**97.14**	94.29	**97.14**
	8	17	3	11
Dermatology	**100**	**100**	72.22	**100**
	6	6	11	16
Spambase	93.06	92.41	88.29	**93.48**
	34	17	18	40
Sonar	76.19	80.95	71.43	**95.23**
	8	9	6	5
Libras Movement	**83.33**	72.22	52.78	80.56
	9	9	4	11
Hill valley with noise	**64.85**	64.19	52.48	59.90
	16	13	2	12
Hill valley without noise	**62.38**	64.03	52.48	60.89
	40	25	2	14
Musk	96.96	97.73	95.15	**98.33**
	26	16	9	42
Musk2	97.45	96.82	96.06	**97.58**
	25	16	13	47
Arrhythmia	**82.22**	68.89	80	**82.22**
	4	8	3	20
Medelon	84.5	83.5	54.83	**84.83**
	18	24	23	12

4.5.5 Comparisons with Previous Research

Lastly, the proposed method (fuzzified CMI+JMBH) is compared against three recent researches on feature selection which are: (Li and Wu, 2008: 224), (Jalali, Nasiri and Minaei, 2009: 721) and (Vieira, Sousa and Kaymak, 2012: 16), using the performance numbers reported in each paper. The results (in Table 4.4) show that the proposed method outperforms (Jalali, Nasiri and Minaei, 2009) and (Li and Wu, 2008) in all common data sets. Comparing with (Vieira, Sousa and Kaymak, 2012), we find that the proposed method gives higher accuracy in 3 out of 4 data sets. Thus, the proposed technique is shown to perform very well across different data sets and in comparison with other techniques.

Table 4.4 Results of The Proposed Method Compared to Those of Other Research Works. Feature Reduction Percentages Relative to The Original Feature Sets are Shown in Parentheses

Dataset	Original Number of Features	(Li and Wu , 2008)	(Jalali, Nasiri and Minaei, 2009)	(Vieira, Sousa and Kaymak, 2012)	Our method
Pima	8	71.51 (89.17)	-	-	**80.52 (50)**
Wine	13	90.99 (83.59)	95.4 (65.38)	96 (69.23)	**100 (76.92)**
Breast Cancer	30	95.97 (96.67)	63.6 (91.66)	98 (86.67)	**98.24 (87.09)**
Sonar	60	68.06 (96.78)	-	86 (86.66)	**95.24 (93.33)**
Arrhythmia	279	-	-	**87 (95.69)**	82.22 (92.83)

4.6 Discussion

The proposed technique employs fuzzification of original features to the filter-based feature selection proposed in chapter 3. The technique allows the feature selection to better handle mixtures of ambiguous and noisy data and improve the selection performance. In this chapter we only apply the proposed method to classification problems. The class attribute can guide the search for features that are related or relevant to the class attribute. For unsupervised feature selection, the class attribute is not available, thus the proposed method needs to be modified. In the next chapter, we present modifications to the method to deal with unsupervised clustering problems.

CHAPTER 5

AN UNSUPERVISED EVOLUTIONARY FUZZY FEATURE SELECTION

Feature selection has been widely studied in supervised classification. However, it is a rather recent and challenging research topic to select a set of predictive features for cluster analysis for two reasons. First, it is not an easy task to define a good criterion to evaluate the quality of a candidate feature set due to the absence of accurate labels of items. Second, feature selection needs to evaluate an exponential number of feature combinations, which is impractical if the data set has a large number of features (Hong, Kwong, Chang and Ren, 2008: 2742).

In this chapter we apply the proposed technique to select features for clustering problem where features are fuzzified using irregular-shaped membership functions evolved by a genetic algorithm that is suitable for high dimensional data.

5.1 Literature Review

As stated in chapter 2, feature selection approaches can be classified into three categories, which are wrapper, filter, and hybrid approaches.

Given an unsupervised clustering problem, a wrapper method incorporates the clustering into the feature subset evaluating process. To evaluate the importance of a candidate feature subset, data clusters are built, validated, and used to evaluate the set. This approach is believed to generate a subset that yields high clustering validity; however, it is likely to take long time.

Hong, Kwong, Chang and Ren (2008: 2744) use population based incremental learning algorithm (PBIL) to generate candidate feature subsets. There are two major steps of the proposed framework. In the first step, the population of clustering solutions is obtained through executing different clustering algorithms and then all the obtained clustering solutions are combined into a single consensus clustering solution.

In the second step, framework searches for a subset of all features that best fits the obtained consensus clustering solution by using PBIL. Similarity measure between each clustering solution is applied to evaluate feature subset.

(Elghazel and Aussem, 2010: 168) extend random forest to unlabeled data by introducing the clustering ensemble to combine data resampling, and random selection of features strategies to generate an ensemble of component clustering. The proposed method randomly selects features and applies a clustering ensemble technique to find a suitable set of clusters. For each cluster, features are selected according to the scree test.

Liu, Liang and Ni (2012: 42) cluster the features and filter out irrelevant or redundant ones from each cluster according to their membership probabilities. New features are obtained by combining features from each different cluster. K-means clustering algorithm is trained on i nstances using these features. The ratio of intra-cluster distance to inter-cluster distance is used to evaluate these clusters, and features with the largest ratio are selected as the final candidate feature sets.

Yang, Hou and Nie (2012: 1794) use K-means clustering to generate class labels and then use joint maximization margin criterion and sparse $L_{2,1}$-norm regularization to perform feature selection. These steps iterate until the algorithm converges to global optimum, and the feature subset, which maximizes the margin between classes, is selected.

A filter-based method generally runs faster and is more preferable for real-world problems, especially those with large data sets. In the filter method, instead of performing clustering as part of the feature selection process, a quality measure is used to evaluate each feature set. This approach composes of two important components, a selection algorithm and a criterion function. A selection algorithm is the candidate feature maker while a criterion function is used to select features and evaluate feature subsets; however, clustering several times is not being performed as the wrapper approach does. Many researchers had found that it yields less effectiveness than the other two approaches; nevertheless, it is not true to state that the filter approach always yields inferior clustering results. Some criterion functions may provide equivalent or better performance than others to guide the search process.

Dash, Choi, Scheuerman and Liu (2002: 117) propose an entropy measure and a forward search to select features. Entropy measure will be low if the data has distinct clusters while the measure will be high if otherwise. This measure is suitable for selecting the most important subset of features because it is invariant with number of dimensions, and is affected only by the quality of clustering. Features that give the minimum entropy for both of intra-cluster and inter-cluster distances are selected.

Liu, Yang, Ding and Ma (2009: 66) use the entropy measure proposed in (Dash, Choi, Scheuerman and Liu, 2002) combined with the Laplacian score (LS) to select features for clustering. Features are ranked according to their LS in descending order. After that, the entropy measure of some features in the top of the list is evaluated and ranked in ascending order. Features with minimum entropy value are selected. This combination solves the drawbacks of using only the Laplacian score.

Suri, Murty and Athithan (2012: 255) apply normalized mutual information (NMI) as an objective function for unsupervised feature selection in an outlier detection problem. Features with high NMI are considered as redundancy thus low NMI features are selected. In their work, average redundancy (AR) of a set of features is defined. One feature at a time is picked to form a feature subset and AR is used to evaluate subset and it is also used as stopping condition of feature selection algorithm.

Liu, Rallo and Cohen (2011: 971) apply a forward search and a kernel least square error (LSE) for unsupervised feature selection. They propose an incremental LSE calculation to evaluate feature subsets in order to improve efficiency. To select a feature, the LSE of unselected features are evaluated. The feature with minimum value of LSE is selected; one feature at a time until stopping condition is reach.

The hybrid approach takes advantage of both wrapper and filter approaches. It applies a filter-based technique to select significant features, and applies a wrapper-based technique to add candidate features and evaluate candidate sets to select the best one. A few works are proposed in this approach for unsupervised learning.

Dash and Liu (2000: 6) apply an entropy measure as a filter criterion. Data in well-formed clusters give low entropy values, so features are ranked in descending order of their entropy values. One feature at a time is picked to form a feature subset.

Each subset is evaluated by K-means clustering, and scattering criteria is used to evaluate cluster quality.

5.2 The Proposed Method

For clustering problem, we have to choose J_1 and J_2 that were not taken into account with the class label. In addition, they must be filter-based unsupervised measure to support cluster analysis.

5.2.1 The First Criterion Function

Criterion functions play significant roles in feature selection. The purpose of the first function is to eliminate redundant features, thus the average redundancy (AR) measure is chosen (Suri, Murty and Athithan, 2012: 255), which can be calculated as follows

$$AR(f_i, F_s) = \frac{1}{|F_s|} \sum_{\forall f_j \in F_s} NMI(f_i, f_j)$$

$$NMI(f_i, f_j) = \frac{I(f_i, f_j)}{\min\{H(f), H(f_j)\}}$$

$$H(x) = -\sum_j p(x_i) \log(p(x_j))$$

$$I(x, y) = \sum_i \sum_j p(x_i, y_j) \log_2 \frac{p(x_i, y_j)}{p(x_i) p(y_j)}$$

where F_s is a set of selected feature, and f_i is the feature to be evaluated.

AR is based on m utual information (MI), which consists of two main properties: the capacity of measuring relationships between variables, and the invariance under space transformations (Kullback, 1997: 22). Therefore, AR is used to select only relevant features while eliminate redundant features that may misguide the search process.

5.2.2 The Second Criterion Function

In classification, feature selection aims at identifying features that predict class labels with the highest accuracy, while clustering feature selection aims at finding those features that discover the natural grouping structures of the data. The entropy theory (Dash, Choi, Scheuerman and Liu, 2002: 116) states that entropy of a system measures the disorder in the system. If the probability of each point is equal and the entropy value is at its maximum, we are most uncertain about the outcome. On the other hand, when the data has well-formed clusters, the uncertainty will be low and so will the entropy (Dash, Choi, Scheuerman and Liu, 2002: 116). Therefore, an entropy measure is chosen as a second criterion function for clustering problems. Entropy measure can be calculated as follows

$$E = \sum_{x_i} \sum_{x_j} E_{ij}$$

$$E_{ij} = \begin{cases} \dfrac{e^{\beta * D_{ij}} - e^{0}}{e^{\beta * \mu} - e^{0}} : 0 \le D_{ij} \le \mu \\ \dfrac{e^{\beta * (1 - D_{ij})} - e^{0}}{e^{\beta * (1 - \mu)} - e^{0}} : \mu \le D_{ij} \le 1 \end{cases}$$

where β is set to 10 from experiments, and μ is calculated as proposed in (Liu, Yang, Ding and Ma, 2009). D_{ij} is the distance between two instances, *i* and *j*. Feature values after fuzzification are nominal, thus the hamming distance (Dash, Liu and Yao, 1997: 535) is used as the distance. D_{ij} is calculated as follows

$$D_{ij} = \frac{\sum_{k=1}^{M} | x_{ik} \ne x_{jk} |}{M}$$

where $| x_{ik} \ne x_{jk} |$ is 0 if x_{ik} equals x_{jk}, and 1, otherwise. M is the number of variables in the subset under consideration. The purpose of using entropy as the second criterion function is to assign low entropy to intra and inter-cluster distances, and to assign higher entropy to noise (Liu, Yang, Ding and Ma, 2009: 66). Therefore, this measure helps the proposed algorithm discover natural groupings of data.

5.2.3 The Proposed Algorithm

The feature to be added must be less redundant than the features that have already been picked. J_1 in forward step is computed according to (5.1).

$$J_{11}(x_n) = AR(x_n, S_{sel}) \text{ where } x_n \in S \setminus S_{sel} \qquad (5.1)$$

In the forward step, J_1 of each unselected feature is evaluated and sorted in an ascending order according to its J_1 values. A candidate set is generated according to (5.2).

$$S^{+}_{cand} = \{x_i \mid x_i \in S \setminus S_{sel} \text{ and } i = [1..d_{cand}] \text{ and } J_{11}(x_1) \leq J_{11}(x_2) \leq \ldots \leq J_{11}(x_{d_{cand}})\} \quad (5.2)$$

A feature to be removed must be one that provides the most redundant, where it produces the largest increase in MI with respect to the remaining $|S_{sel}|$ - 1 features. Therefore, J_1 in backward step is computed according to (5.3).

$$J_{12}(x_n) = \sum_{n=1, n \neq S}^{|S_{sel}|} NMI(x_n, x_s) \text{ where } x_n \in S_{sel}, x_s = S_{sel} \setminus x_n \quad (5.3)$$

In backward step, J_1 of each selected feature is evaluated and sorted in a descending order. A candidate set is generated according to (5.4).

$$S^{-}_{cand} = \{x_i \mid x_i \in S_{sel} \text{ and } i = [1..d_{cand}] \text{ and } J_{12}(x_1) \geq J_{12}(x_2) \geq \ldots \geq J_{12}(x_{d\,cand})\} \quad (5.4)$$

5.3 K-Means Clustering

K-means clustering is an unsupervised learning algorithm. It is the most widely used technique to cluster data (Kasliwal and Lade, 2013: 981). It is simple but efficient to partition data into clusters. The algorithm is an iterative process to group data into K clusters, where the sum of within-cluster distances between point-to-cluster centroid over all clusters is minimized. In this work, we apply the squared Euclidean distance to measure the distance between a point and its centroid of cluster, which is the mean value of the points in that cluster. Squared Euclidean distance is evaluated as follows.

$$d = \sum_{i=1}^{n} (x_i - c)^2$$

where x_i is a data instance, and *c* is the group centroid.

The K-means algorithm starts by randomly select K representative data from the raw data to be the centroid for each of the K data groups. Then, it assigns each data to the closet group based on the distance measure. The algorithm updates the centroid value of the group for the mean of the data in that group, and repeatedly reassigns groups and updates centroids until the assignment cannot be changed.

5.4 Experimental Results

To evaluate the effectiveness of our method, we use the evaluation method previously used in Dy and Brodley (2004); Zhao, Kwok, Wang and Zhang (2009); Elghazel and Aussem (2013) and Markov (2013) where data sets with predefined classes are used. The data sets used in the evaluations of the proposed technique consists of sixteen data sets from the UCI machine learning repository.

5.4.1 Data Sets

Data used in the experiments are fifteen standard data sets with various sizes from the UCI machine learning repository (Asuncion and Newman, 2007). The detail of all data sets is shown in Table 5.1.

5.4.2 Experimental Setup

After performing clustering in each data set, each cluster will be assigned a label of class majority. The total number of correctly labeled data divided by the total number of instances in the data equals the accuracy of clustering. Since the K-means algorithm is sensitive to the initialization of cluster centroids and the value of K, K is set as the number of classes, and the final accuracy is obtained from an average value after twenty runs.

After exploring various combinations of system parameters, the followings are used in all experiments. For the hierarchical co-evolutionary genetic algorithm, the population sizes of the first and second layer are set to 100 and 30, respectively, upper bound value = 0.8, lower bound = 0.1. Uniform mutation operator is used for both layers, and the candidate set size for the feature selection is $d_{cand} = 5$. For the data set without a separate test set provided, a 10-fold cross validation is used to measure the performance.

Table 5.1 Data Sets Used in Experiment

Name	Feature Type	Number of instances	Number of features
Iris	Numeric	150	4
EColi	Numeric	336	7
Pima	Numeric	768	7
Wine	Numeric	178	13
Image Segmentation	Numeric	2310	19
Parkinson	Numeric	195	22
Breast Cancer	Numeric	569	30
Ionosphere	Numeric	351	34
Dermatology	Numeric	366	34
Spambase	Numeric	4601	57
Sonar	Numeric	208	60
Libras Movement	Numeric	360	90
Hill valley with noise	Numeric	1212	100
Arrhythmia	Numeric	452	279
Madelon	Numeric	2600	500

5.4.3 Effectiveness of the Proposed Technique

In this section, we studied the benefit of feature selection in comparison to the original set of features and effectiveness of the proposed two-criterion function feature selection algorithm. We also studied three feature selection approaches, the original filter-based, non-fuzzified two-criterion (J_1+J_2 without fuzzy), and fuzzified two-criterion (J_1+J_2 with fuzzy). The results (in Table 5.2) show that feature selection is beneficial and gives higher accuracy than the original set of features. The proposed feature selection algorithm is effective; nevertheless, nine out of fifteen cases of J_1+J_2 without fuzzy yields better results than J_2 alone, and thirteen out of fifteen cases of fuzzified J_1+J_2 gives better results than the best of J_2 and J_1+J_2 without fuzzy configurations, except for Spambase and Madelon data sets. For

Madelon, the proposed method gives slightly lower accuracy (0.16%) with twenty features less than the best method.

5.4.4 Comparisons with Previous Research

Lastly, the proposed method is compared with three recent researches on feature selection for clustering, which are (Hong, Kwong, Chang and Ren, 2008; Elghazel and Aussem, 2010 and Liu, Liang and Ni, 2012) using common data sets and performance numbers reported in each paper. All of these methods used the wrapper approach where clustering are performed to evaluate each feature subset. Results are shown in Table 5.3. We can see that our method outperforms (Hong, Kwong, Chang and Ren, 2008) in four out of five data sets. Comparing with (Elghazel and Aussem, 2010), our method outperforms in Iris data set but performs worse on the Ecoli data set.

However, the proposed method gives higher performance than (Liu, Liang and Ni, 2012) in two out of three common data sets. This is due to the fact that our technique uses the filter approach while the referenced technique uses the wrapper approach, which takes much more time.

Table 5.2 The Results of The Original Filter Method (J_2), J_1+J_2 Without Fuzzification, and J_1+J_2 with Fuzzification

Data set		**Original data**	J_2	J_1+J_2 **Without Fuzzy**	J_1+J_2 **With Fuzzy**
Iris	Accuracy	91.67	91.37	91.37	**96**
	Features	4	2	2	1
EColi	Accuracy	80.54	79.87	81.41	**82.07**
	Features	7	6	6	6
Pima	Accuracy	66.02	73.18	66.02	**74.22**
	Features	7	5	5	5
Wine	Accuracy	72.72	76.79	68.82	**89.89**
	Features	13	4	8	2
Image Segmentation	Accuracy	32.68	56.83	62.45	**65.64**
	Features	19	16	15	12
Parkinson	Accuracy	75.39	75.39	75.39	**76.61**
	Features	22	1	1	5
Breast Cancer	Accuracy	84.28	85.41	89.63	**90.33**
	Features	30	16	9	2
Dermatology	Accuracy	35.70	79.75	85.85	**86.31**
	Features	34	29	17	23
Ionosphere	Accuracy	71.22	74.39	74.93	**77.35**
	Features	34	1	1	2
Spambase	Accuracy	60.81	62.23	**68.74**	66.98
	Features	57	21	40	3
Sonar	Accuracy	57.89	59.93	57.57	**67.86**
	Features	60	37	14	11
Movement Libras	Accuracy	40.63	43.24	45.71	**46.01**
	Features	90	75	50	46
Hill valley with noise	Accuracy	50.55	50.66	**50.83**	**50.83**
	Features	100	1	4	1
Arrhythmia	Accuracy	58.81	58.50	58.68	**58.85**
	Features	279	40	27	42
Medelon	Accuracy	55.89	51.81	**62.83**	62.67
	Features	500	10	34	14

Table 5.3 Experimental Results of The Proposed Method Compared with Previously Proposed Methods. The Number of Selected Features Is Shown in The Parenthesis.

Dataset	**Original Number of Features**	**(Hong, Kwong, Chang and Ren, 2008)**	**(Elghazel and Aussem, 2010)**	**(Liu, Liang and Ni, 2012)**	**Proposed Method**
Iris	4	92.56 (2)	93.2 (1.78)	92.53 (2)	**96 (1)**
EColi	7	82.51 (6)	**84.02 (5.3)**	81.73 (4)	82.07 (6)
Wine	13	87.07 (6)	-	**94.46 (4)**	89.89 (2)
Breast Cancer	30	77.86 (12)	-	-	**90.33 (2)**
Ionosphere	34	59.17 (19)	-	-	**77.35(2)**

5.5 Discussion

This chapter applies the proposed technique to select a minimal set of features for clustering problems. The first function is applied to eliminate features with redundant effects, and the second function is used to select a feature subset that yields well-formed clusters. The technique is evaluated using sixteen standard UCI data sets and is compared with three recent research papers. The results show that feature selection is beneficial to clustering, the two-criterion feature selection algorithm is generally effective, and that the fuzzification improves the performance of the feature selection algorithm. In addition, the proposed technique performs well in comparison with the wrapper-based feature selection methods previously proposed in common data sets.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

6.1 Conclusions

The objective of this dissertation is to improve effectiveness of feature selection that can improve effectiveness of data mining algorithms. We want to enhance feature selection performance while, at the same time, make it as efficient as possible, especially with large data sets.

Contributions of this work are summarized as follows,

1) A filter-based feature selection using two criterion functions was proposed. Filter based has been accepted for low computation time with moderate accuracy; however, criterion functions have different characteristics. We combined two criterion functions together to utilize their advantages in improving data mining algorithm accuracy. The results of the experiments show that the proposed method gives higher accuracy to the selected features than the entire set of features and the proposed method also outperforms other feature selection methods; moreover, the method does not take much computation time.

2) A feature fuzzification to the two-stage feature selection algorithm was introduced to handle noisy and distraction within the data. Fuzzy logic needs membership functions to transform crisp data into fuzzy values before passing them through the feature selection process. Irregular-shaped membership function is used instead of predefined regular shapes by experts. An irregular-shape is flexible, thus it can be fitted into actual distribution of the data set. The fuzzy logic enhances feature selection effectiveness and improves classification accuracy by outperforming other recent researches.

3) Not only in the supervised learning that the proposed method can be applied to, it can also be applied to the unsupervised clustering context. Two proper criterion functions are used for features and subset evaluation for the unsupervised clustering. Experimental results show that the proposed method can be applied to unsupervised clustering and helps improve clustering accuracy.

6.2 Future Work

Genetic algorithm is the learning technique that we use to generate irregular-shaped membership functions. Although we use criterion functions to evaluate an individual chromosome for the GA, the learning process still takes time. In addition, it requires many parameters to search for an optimal solution. Some issues can be further studied.

1) Number of criterion functions can be more than two. We may apply larger number of simple criterion functions instead of using two complicated criterion functions; however applied criterion functions must be consistent and they should be suitable for the problem. Number of candidate features must decrease continuously from the first criterion function until the best one is obtained from the last criterion function.

2) Self-adaptive learning or other optimization techniques can be applied to adjust the points of irregular-shaped membership function. As we had shown in the experiments, fuzzy logic combined with GA can improve feature selection effectiveness; however, it takes time to iteratively adjust points of irregular-shaped MF, encoded in chromosomes. Other self-adaptive learning or optimization techniques techniques may converge to optimum solution more rapidly and still give high performance of feature selection process.

3) In this dissertation, we employ fuzzy logic as a preprocessing step to refine quality of data. Some research had stated that, information loss could happen if preprocessing step is not sturdy enough. Using a fuzzy measure as a criterion function may relieve this problem because original data is used in feature selection algorithm instead of using fuzzy value. A fuzzy measure can handle noise or distractions, and if the fuzzy measure is suitable, the effectiveness of selected features will be improved.

Moreover, it could reduce number of selected features and time required of feature selection process.

BIBLIOGRAPHY

Ansari, Zahid.; Azeem, M.; Babu, Vinaya and Ahmed, Waseem. 2012. A Fuzzy Approach for Feature Evaluation and Dimensionality Reduction to Improve the Quality of Web Usage Mining Results. **International Journal on Advanced Science Engineering Information Technology.** 2, 6 (June): 67-73.

Asuncion, A. and Newman, D. 2007. **UCI Machine Learning Repository**. Retrieved January 25, 2011 from http://archive.ics.uci.edu/ml

Breiman, L.; Friedman, J. H.; Olshen, R. A. and Stone, C. J. 1984. **Classification and Regression Trees**. Pacific Grove, CA: Wadsworht.

Bruzzone, Lorenzo.; Roli, Fabio and Serpico, Sebastiano B. 1995. An Extension of the Jeffreys-Matusita Distance to Multiclass Cases for Feature Selection. **IEEE Transaction on GeoScience and Remote Sensing.** 33, 6 (November): 1318-1321.

Bruzzone, Lorenzo and Serpico, Sebastiano B. 2000. A Technique for Feature Selection in Multiclass Problems. **International Journal of Remote Sensing.** 21, 3 (February): 549-563.

Cai, Deng.; Zhang, Chiyung and He, Xiaofei. 2010. Unsupervised Feature Selection for Multi-cluster Data. In **Proceeding KDD'10 Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.** New York, USA: ACM. Pp. 333-342.

Cintra, Marcos E.; Martin, Trevor P.; Monard, Maria C. and Camargo, Heloisa A. 2009. Feature Subset Selection Using a Fuzzy Method. In **International Conference on Intelligent Human-Machine Systems and Cybernetics**. New York, USA: IEEE. Pp. 214-217.

Dash, Manoranjan.; Choi, Kiseok.; Scheuerman, Peter and Liu, Huan. 2002. Feature Selection for Clustering-A Filter Solution. In **Proceedings of International Conference on Data Mining**. New York, USA: IEEE. Pp. 115-122.

Dash, Manoranjan.; Liu, Huan and Yao, Jun. 1997. Dimensionality Reduction for Unsupervised Data. In **Proceedings of Ninth IEEE International Conference Tools with Artificial Intelligence.** New York, USA: IEEE. Pp. 532-539.

Dash, Manoranjan and Liu, Huan. 2000. Feature Selection for Clustering. In **Proceedings of Fourth Pacific-Asia Conference on Knowledge Discovery and Data Mining.** Düsseldorf, Germany**:** Springer Berlin Heidelberg. Pp. 110-121.

De Maesschalck, Roy.; Jouan-Rimbaud, Delphine and Massart, Désiré L. 2000. The Mahalanobis Distance. **Chemometrics and Intelligent Laboratory Systems.** 50, 1 (January): 1-18.

Dy, Jennifer G. and Brodley, Carla E. 2004. Feature Selection for Unsupervised Learning. **Journal of Machine Learning Research.** 5, 1 (January): 845-889.

Elghazel, Haytham and Aussem, Alex. 2010. Feature Selection for Unsupervised Learning using Random Cluster Ensembles. In **IEEE International Conference on Data Mining.** New York, USA: IEEE. Pp. 168-175.

Elghazel, Haytham and Aussem, Alex. 2013. Unsupervised Feature Selection with Ensemble Learning. **Machine Learning.** (April): 1-24.

Engelbrecht, Andries P. 2007. **Computational Intelligence: An Introduction**. 2nd ed. Chichester, England: John Wiley and Sons Ltd.

Estévez, Pablo A.; Tesmer, Michel.; Perez Claudio A. and Zurada, Jacek M. 2009. Normalized Mutual Information Feature Selection. **IEEE Transaction on Neural Networks.** 20, 2 (February): 189-201.

Fleuret, François. 2004. Fast Binary Feature Selection with Conditional Mutual Information. **Journal of Machine Learning Research.** 5, 11 (November): 1531-1555.

Gan, John Q.; Hasan, Bashar Awwad Shiekh and Tsui, Chun Sing louis. 2011. A Hybrid Approach to Feature Subset Selection for Brain-Computer Interface Design. In **IDEAL 2011, LNCS 6936.** Düsseldorf, Germany**:** Springer Berlin Heidelberg. Pp. 279-286.

Grande, Javier.; Suárez, María del Rosario and Villar, José Ramón. 2007. A Feature Selection Method Using a Fuzzy Mutual Information Measure. In **Innovations in Hybrid Intelligent Systems, ASC 44**. Düsseldorf, Germany**:** Springer Berlin Heidelberg. Pp. 56-63.

Guan, Yue.; Dy, Jennifer G. and Jordan, Michael I. 2011. A Unified Probabilistic Model for Global and Local Unsupervised Feature Selection. In **Proceedings of the 28th International Conference on Machine Learning.** Bellevue, USA: ICML. Pp. 1073-1080.

Haindl, Michal.; Somol, Petr.; Ververidis, Dimitrios and Kotropoulos, Constantine. 2006. Feature Selection Based on Mutual Correlation. In **Progress in Pattern Recognition, Image Analysis and Application, Lecture Notes in Computer Science.** Düsseldorf, Germany**:** Springer Berlin Heidelberg. Pp. 569-577.

Hedjazi, Lyamine.; Kempowsky-Hamon, Tatiana.; Despènes, Laurène.; Le Lann, Marie-Véronique.; Elgue, Sébastien and Aguilar-Martin, Joseph. 2010. Sensor Placement and Fault Detection Using an Efficient Fuzzy Feature Selection Approach. In **49th IEEE Conference on Decision and Control**. New York, USA: IEEE. Pp. 6827-6832.

Hong, Yi.; Kwong, Sam.; Chang, Yuchou and Ren, Qingsheng. 2008. Unsupervised Feature Selection Using Clustering Ensembles and Population Based Incremental Learning Algorithm. **Pattern Recognition.** 41, 9 (September): 2742-2756.

Huang, Haoming.; Pasquier, Michel and Quek, Chai. 2007. HiCEFS-A Hierarchical Coevolutionary Approach for the Dynamic Generation of Fuzzy System. In **IEEE Congress on Evolutionary Computation, CEC**. New York, USA: IEEE. Pp. 3426-3443.

Jalali, Laleh.; Nasiri, Mahdi and Minaei, Behrooz. 2009. A Hybrid Feature Selection Method Based on Fuzzy Feature Selection and Consistency Measures. In **Intelligent Computing and Intelligent System (ICIS)**. New York, USA: IEEE. Pp. 718-722.

Kasliwal, Niraj N. and Lade, Shrikant. 2013. Clustering of Datasets by Using K-Means & C-Means (Fuzzy) Methodology. **International Journal of Engineering Research & Technology.** 2, 4 (April): 980-987.

Kullback, S. 1997. **Information Theory and Statistic**. New York: Dover.

Leng, Jinsong.; Valli, Craig and Armstrong, Leisa. 2010. A Wrapper Based Feature Selection for Analysis of Large Data Sets. In **Proceedings of 2010 3rd International Conference on Computer and Electrical Engineering.** New York, USA: IEEE. Pp. 167-170.

Li, Yun and Wu, Zhong F. 2008. Fuzzy Feature Selection Based on Min-Max Learning Rule and Extension Matrix. **Pattern Recognition.** 41, 1 (January): 217-226.

Liu, Huan and Yu, Lei. 2005. Toward Integrating Feature Selection Algorithms for Classification and Clustering. **Knowledge and Data Engineering, IEEE Transaction.** 17, 4 (April): 491-502.

Liu, Rong.; Rallo, Robert and Cohen, Yoram. 2011. Unsupervised Feature Selection Using Incremental Least Squares. **International Journal of Information Technology & Decision Making.** 10, 6 (November): 967-987.

Liu, Rongye.; Yang, Ning.; Ding, Xiangqian and Ma, Lintao. 2009. Unsupervised Feature Selection Algorithm: Laplacian Score Combined with Distance-Based Entropy Measure. In **Third International Symposium on Intelligent Information Technology Application.** New York, USA: IEEE. Pp. 65-68.

Liu, Tong.; Liang, Yongquan and Ni, Weijian. 2012. A Novel Approach to Feature Selection for Clustering. In **Fifth International Conference on Intelligent Computation Technology and Automation**. New York, USA: IEEE. Pp. 41-44.

Markov, Zdravko. 2013. MDL-Based Unsupervised Attribute Ranking. In **Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference.** California, USA: AAAI. Pp. 444-449.

Parthaláin, Neil M. Jr. and Jensen, Richard. 2010. Measures for Unsupervised Fuzzy-Rough Feature Selection. **International Journal of Hybrid Intelligent Systems.** 7, 4 (December): 249-259.

Pereira, Roberto R. Jr. et al. 2007. Usefulness of Texture Analysis for Computerized Classification of Breast Lesions on Mammograms. **Journal of Digital Imaging.** 20, 3 (September): 248-255.

Pudil, Pavel.; Novovičová, Jana and Kittler, Josef. 1994. Floating Search Methods in Feature Selection. **Pattern Recognition Letters.** 15, 11 (November): 1119-1125.

Sánchez-Maroño, Noelia S.; Alonso-Betanzos, Amparo A. and Castillo, Enrique. 2005. A New Wrapper Method for Feature Subset Selection. In **Proceedings-European Symposium on Aritificial Neural Networks.** New York, USA: IEEE. Pp. 515-520.

Somol, Petr.; Novovičová, Jana and Pudil, Pavel. 2006. Flexible-Hybrid Sequential Floating Search in Statistical Feature Selection. **Structural, Syntactic, and Statistical Pattern Recognition, Lecture Notes in Computer Science**. 4109 (August): 632-639.

Somol, Petr.; Pudil, Pavel.; Novovičová, Jana and Pacliík, Pavel. 1999. Adaptive Floating Search Methods in Feature Selection. **Pattern Recognition Letters.** 20, 11 (November): 1157-1163.

Songyot Nakariyakul and Casasent, David P. 2009. An Improvement on Floating Search Algorithm for Feature Subset Selection. **Pattern Recognition.** 42, 9 (September): 1932-1940.

Sun, Hongbin.; Wang, Hao.; Zhang, Boming and Zhao, Feng. 2010. PGFB : A Hybrid Feature Selection Method Based on Mutual Information. In **Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2010)**. New York, USA: IEEE. Pp. 2862-2871.

Suri, Ranga N. N. R.; Murty, Narasimha M. and Athithan, G. 2012. Unsupervised Feature Selection for Outlier Detection in Categorical Data Using Mutual Information. In **12th International Conference on Hybrid Intelligent System (HIS)**. New York, USA: IEEE. Pp. 253-258.

Tsai, Cheng-Jung.; Lee, Chien-I and Tang, Wei-Pang. 2008. A Discretization Algorithm Based on Class-Attribute Contingency Coefficient. **Information Sciences.** 178, 3 (February): 714-731.

Vieira, Susana M.; Sousa, João M.C. and Kaymak, Uzay. 2012. Fuzzy Criteria for Feature Selection**. Fuzzy Sets and Systems.** 189, 1 (February): 1-18.

Yang, Shizhun.; Hou, Chenping and Nie, Feiping. 2012. Unsupervised Maximum Margin Feature Selection via $L_{2,1}$-Norm Minimization. **Neural Computing and Application**. 21, 7 (October): 1791-1799.

Yu, Lei and Liu, Huan. 2003. Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. In **Proceedings of the Twentieth International Conference on Machine Learning (ICML)**. Bellevue, USA: ICML. Pp. 856-863

Zhang, Li-Xin.; Wang, Jia-Xin.; Zhao, Yan-Nan and Yang, Ze-Hong. 2003. A novel Hybrid Feature Selection Algorithm:Using RELIEFF Estimation for GA-Wrapper Search. In **Proceedings of the Second International Conference on Machine Learning and Cybernetics**. New York, USA: IEEE. Pp. 380-384.

Zhao, Bin.; Kwok, James.; Wang, Fei and Zhang, Changshui. 2009. Unsupervised Maximum Margin Feature Selection with Manifold Regularization. In **Computer Vision and Pattern Recognition, CVPR**. New York, USA: IEEE. Pp. 888-895.

Zhou, Yaqian.; Weng, Fuliang.; Wu, Lide and Schmidt, Hauke. 2003. A Fast Algorithm for Feature Selection in Conditional Maximum Entropy Modeling. In **Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing**. New York, USA: IEEE. Pp. 153-159.

Zhuo, Li.; Zheng, Jing.; Wang, Fang.; Li, Xia.; Ai, Bin and Qian, Junping. 2008. A Genetic Algorithm Based Wrapper Feature Selection Method for Classification of Hyperspectral Images Using Support Vector Machine. **The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**. XXXVII (Part B7): 397-402.

BIOGRAPHY

Name Jitwadee Chaiyakarn

ACADEMIC BACKGROUND Bachelor's Degree with major in Electrical Engineering from Kasetsart University, Bangkok, Thailand in 2000 and a Master's Degree in Computer Science at National Institute of Development Administration, Bangkok, Thailand in 2004

EXPERIENCES Senior programmer of Phatra insurance Co. Ltd in 2004 - 2005
Chief of Computer Application Developer of Phatra insurance Co. Ltd in 2006 - 2009
Senior Manager at LoxbitPA Public, Co. Ltd in 2009 - 2011