# AN INTERPRETABLE FUZZY-ENSEMBLE METHOD FOR CLASSIFICATION AND DATA ANALYSIS

ADEL LAHSASNA

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2016

# AN INTERPRETABLE FUZZY-ENSEMBLE METHOD FOR CLASSIFICATION AND DATA ANALYSIS

ADEL LAHSASNA

THESIS SUBMITTED IN FULFILMENT
OF THE REQUIREMENTS
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND
INFORMATION TECHNOLOGY

UNIVERSITY OF MALAYA
KUALA LUMPUR

2016

# UNIVERSITI MALAYA

## ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: **ADEL LAHSASNA**

Registration/Matric No: **WHA090025**

Name of Degree: **Ph.D. in Computer Science**

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"):

**AN INTERPRETABLE FUZZY-ENSEMBLE METHOD FOR CLASSIFICATION AND DATA ANALYSIS**

Field of Study: **Artificial Intelligence - Soft Computing Techniques**

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date

Subscribed and solemnly declared before,

Witness's Signature Date

Name:
Designation:

## ABSTRACT

Despite the advantage of being highly accurate classifiers, many machine learning methods such as artificial neural networks (ANNs) and support vector machines (SVMs) have been criticized for their lack of interpretability as users are prevented from knowing about the decision process of their inner systems. Interpretability, which refers to the ability of a system to express its behavior in an understandable way, has recently gained more attention and it is considered as an important requirement especially for those applications that use knowledge-based systems such as decision support systems. The main objective of our study is to propose an interpretable fuzzy-ensemble method that can be used for both classification and data analysis. This classifier is the result of combining the advantages of an interpretable fuzzy rule-based system and accurate ensemble method. To achieve the aforementioned objective, we firstly propose two variant methods of a well-known fuzzy classifier proposed in (Ishibuchi & Nojima, 2007) aiming to improve its ability to maximize the accuracy of the fuzzy rule-based system while preserve its interpretability. In addition, we proposed a feature selection-based method that aims to improve the quality of the non-dominated fuzzy rule-based systems especially those generated from high dimensional data sets by allowing the genetic algorithm (GA) to start from a good initial population.

For the ensemble method, we propose a design that combines five different base classifiers and use a GA-based selection method to select a subset from all the ensemble outputs using accuracy and diversity measures as two objectives in the fitness function. In addition, we propose a combination method that aims to improve the accuracy of the fuzzy rule-based system by using the accurate ensemble method to classify the patterns that have low certainty degree or in cases of rejected and uncovered classifications.

The proposed method is tested using six data sets from the UCI machine learning repository, and the obtained results are compared with other benchmark methods. The results show that the fuzzy-ensemble method was able to maintain to a great extent the superiority of the ensemble method accuracy over the fuzzy rule-based system by successfully retaining an average of 76.77% of the accuracy gains obtained by the ensemble method relative to fuzzy rule-based system. In addition, the fuzzy-ensemble method has successfully preserved its interpretability compared to the fuzzy rule-based system. In addition, the two developed methods, namely, the fuzzy rule-based system and the ensemble method have shown separately competitive results with their related methods proposed in the literature. Thus, in addition to the proposed fuzzy-ensemble method, they can be separately used as single classifiers.

## ABSTRAK

Meskipun mempunyai kelebihan sebagai kaedah pengkelasan yang sangat tepat, kebanyakan kaedah pembelajaran mesin seperti rangkaian pembuatan neural (ANN) dan mesin sokongan vektor (SVMs) telah dikritik kerana kelemahan menginterpretasi oleh pengguna telah dihalang daripada mengetahui tentang keputusan proses dalaman sistem. Menginterpretasi, yang merujuk kepada keupayaan sistem untuk menyatakan perilaku dengan cara yang mudah difahami, perkara ini telah mendapat lebih perhatian pada masa kini dan ia dianggap sebagai satu keperluan yang penting untuk kebanyakan aplikasi terutama bagi mereka yang menggunakan sistem berasaskan pengetahuan seperti sistem sokongan keputusan. Objektif utama kajian kami adalah mengemukakan satu kaedah yang dinamakan pertunjukkan-samar yang boleh digunakan untuk klasifikasi dan data analisis. Pengkelas ini adalah hasil daripada gabungan kelebihan sistem berasaskan kaedah peraturan penginterpretasi samar dan kaedah pertunjukkan tepat. Untuk mencapai matlamat seperti yang dinyatakan di atas, pertama sekali, kami mengemukakan dua varian yang umumnya di ketahui iaitu pengkelas kabur yang telah dikemukan dalam (Ishibuchi & Nojima 2007) yang bertujuan untuk meningkatkan keupayaan untuk memaksimumkan ketepatan sistem berasaskan peraturan samar dan memelihara keupayaan penginterpreatsinya. Selain itu, kami mengemukakan satu kaedah berasaskan pemilihan ciri yang bertujuan untuk meningkatkan kualiti sistem yang tidak didominasi oleh peraturan berasaskan kaedah samar khususnya set-set data yang dijanakan daripada dimensi tinggi dengan membenarkan algoritma GA bermula dari populasi awal yang baik.

Mengenai kaedah pertunjukkan, kami mencadangkan satu reka bentuk yang menggabungkan lima pengkelas yang berlainan dan menggunakan kaedah pemilihan

berasaskan GA untuk memilih subset dari semua hasil pertunjukkan dengan menggunakan ketepatan dan kepelbagaian langkah sebagai dua objektif dalam fungsi kecergasan. Selain itu, kami mengemukakan satu kaedah gabungan yang bertujuan untuk meningkatkan ketepatan sistem berasaskan peraturan samar dengan menggunakan kaedah pertunjukkan yang tepat untuk mengkelaskan corak yang mempunyai darjah kepastian yang rendah atau dalam kes-kes klasifikasi yang tidak diterima dan tidak dilindungi.

Kaedah yang dikemukakan diuji dengan menggunakan enam set data daripada mesin pembelajaran repositori UCI, dan keputusan yang diperolehi dibandingkan dengan kaedah penanda aras lain. Hasil kajian menunjukkan bahawa kaedah pertunjukkan-samar dapat mengekalkan tahap yang mengagumkan iaitu keunggulan ketepatan bagi kaedah pertunjukkan mengatasi sistem berasaskan peraturan samar dan berjaya mengekalkan purata 76,77% daripada ketepatan yang diperolehi dengan menggunakan kaedah pertunjukkan relatif sistem yang berasaskan kaedah samar. Di samping itu, kaedah pertunjukkan-samar telah berjaya memelihara interpretasi berbanding dengan sistem berasaskan peraturan samar. Tambahan pula, kedua-dua kaedah yang telah dibangunkan ini, iaitu, sistem yang berasaskan peraturan samar dan kaedah pertunjukkan telah menunjukkan keputusan berasingan yang kompetitif dengan menggunakan kaedah yang berkenaan. Oleh itu, sebagai tambahan kepada kaedah pertunjukkan-samar seperti yang dicadangkan, ia-nya boleh digunakan secara berasingan sebagai pengkelas tunggal.

## ACKNOWLEDGEMENT

Alhamdulillah, all praises to Allah for giving me the strength to complete this thesis. I owe my deepest gratitude to my supervisor Dr. Woo Chaw Seng for his advice, guidance, and patience throughout this study. This research will never be accomplished without his supervision. I'd like also to extend my gratitude to my former supervisors, Prof. Dr. Roziati Zainuddin and Prof Datin Raja Noor Ainon for their deep commitments and continued help and support.

I would like to express my warmest thanks to the staffs of the Faculty of Computer Science and Information Technology (FCSIT), University Malaya (UM) for their help and support. I am also grateful to the University of Malaya for giving me the opportunity to further my study at FCSIT, UM.

I am indebted to many of my colleagues, for the words of courage and support in various possible ways.

I dedicated this thesis to my late father and mother for all the prayers that help me to go through this lonely journey of research. My special thanks to my wife Amel Bezziche for her patience and encouragement and to my lovely daughters: Douaa, Iman and Aya.

**TABLE OF CONTENTS**

## LIST OF FIGURES

## LIST OF TABLES

## LIST OF ABBREVIATIONS AND ACRONYMS

| **Abbreviation** | **Meaning** |
|---|---|
| 10-CV | **10**-fold **C**ross-**V**alidation |
| Adaboost | **Ada**ptive **Boost**ing |
| ANFIS | **A**daptive **N**euro-**F**uzzy **I**nference **S**ystem |
| ANNs | **A**rtificial **N**eural **N**etwork**s** |
| Bagging | **B**ootstrap **agg**regat**ing** |
| CART | **C**lassification **A**nd **R**egression **T**ree |
| CIFE | **C**onditional **I**nformative **F**eature **E**xtraction |
| CMIM | **C**onditional **M**utual **I**nfo **M**aximisation |
| CONDRED | **Cond**itional **Red**undancy |
| DB | **D**ata **B**ase |
| DF | **D**ouble De**f**ault |
| DIFF | **Diff**iculty |
| DISR | **D**ouble **I**nput **S**ymmetrical **R**elevance |
| EFuNN | **E**volving **Fu**zzy **N**eural **N**etworks |
| FALCON | **F**uzzy **A**daptive **L**earning **Co**ntrol **N**etwork |
| FINEST | **F**uzzy **In**ference **E**nvironment **S**oftware With **T**uning |
| FRBSs | **F**uzzy **R**ule-**B**ased **S**ystem**s** |
| FUN | **Fu**zzy **N**et |
| FURIA | **F**uzzy **U**nordered **R**ule **I**nduction **A**lgorithm |
| GARIC | **G**eneralized **A**pproximate **R**easoning-Based **I**ntelligent **C**ontrol |
| GAs | **G**enetic **A**lgorithm**s** |
| ICAP | **I**nteraction **Cap**ping |
| JMI | **J**oint **M**utual **I**nformation |
| LDA | **L**inear **D**iscriminant **A**nalysis |
| MFs | **M**embership **F**unction**s** |
| MIM | **M**utual **I**nformation **M**aximisation |
| MOEAs | **M**ulti-**O**bjective **E**volutionary **A**lgorithm**s** |
| NB | **N**aïve **B**ayes |
| NEFCLASS | **Ne**uro-**F**uzzy **Class**ification |
| NEFCON | **Ne**uro-**F**uzzy **Con**trol |
| NSGA-II | **N**on-Dominated **S**orting **G**enetic **A**lgorithm-II |
| RB | **R**ule **B**ase |
| RF | **R**andom **F**orest |
| RLO | **R**andom **L**inear **O**racle |
| RS | **R**andom **S**ubspace |
| RSO | **R**andom **S**phere **O**racle |
| SLAVE | **S**tructural **L**earning **A**lgorithm On **V**ague **E**nvironment |
| SONFIN | **S**elf-C**o**nstructing **N**eural **F**uzzy **I**nference **N**etwork |
| SVMs | **S**upport **V**ector **M**achine**s** |

## LIST OF APPENDICES

# CHAPTER 1: INTRODUCTION

## 1.1 Background

Over the last few decades, there have been tremendous improvements in methods and computer hardware capacities for collecting and storing large volumes of data. In response to this dramatic increase of stored data, data analysts and decision makers in business companies and other organizations have been eagerly looking for tools and software that can help them making sense of their collected data. This increasingly interest in extracting useful information from raw data has boosted the research in data mining and machine learning fields which aim to develop more effective algorithms that can be used for building models for the purpose of prediction and knowledge discovery (Liao, Chu, & Hsiao, 2012).

Several machine learning methods such as Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs) have been successfully applied in many real-world classification problems from different disciplines such as business, finance, medical, etc. Their use to model classification problems is motivated mainly by their ability to build highly accurate models. Despite this advantage, these methods have been criticized for their lack of interpretability or transparency as they are considered as black box systems, i.e. users are prevented from knowing about the decision process of their inner systems (Baesens, Setiono, Mues, & Vanthienen, 2003).

The interpretability in a system, which is generally defined as the ability of this system to express its behavior in an understandable way, is a desirable property for all kinds of applications or systems and it is even an essential requirement for those using knowledge-based systems such as decision support systems applied in medicine, finance, and other domains (J. M. Alonso & Magdalena, 2011b). The lack of this property is a drawback that may limit or prevent the use of these methods in these kinds of applications. One advantage of an interpretable system, which used for example in a medical diagnosis application, is its ability to give a response on how a particular decision of diagnosis was made. This feature is of a great importance as it may allow the physician to either consider or reject the decision made by a given diagnosis system. In addition, it may help to understand or even reveal some relations between the signs exhibited by the patients and the diagnosis outcome.

As it is stated earlier, there is a growing interest to turn the raw data into useful information and interpretable systems can play an important role as they allow for representing the extracted or induced knowledge from data in a way that is understandable by human beings.

Unlike ANNs and SVMs, fuzzy rule-based systems offer a convenient format for representing the knowledge underlying a classification system in the form of transparent and linguistic conditional statements (Dubois & Prade, 1996). These statements are in the form of “if condition(s) X is (are) met then class is Y”. Such a format is humanly understandable as it uses a language close to the natural language used by human beings (Fernández, López, del Jesus, & Herrera, 2015). In addition, fuzzy rule-based systems allow for the incorporation of both expert knowledge and knowledge extracted from data in one knowledge base system (J. M. Alonso, 2007).

Furthermore, fuzzy rule-based systems are universal approximators as they have the ability to learn non-linear mappings between inputs and outputs (Serge Guillaume, 2001). This ability has been widely employed to build models that simulate the behavior of many real-world systems including classification systems. The twofold identity of fuzzy rule-based systems, namely, extracting classifiers from data sets and the use of linguistic variables for knowledge representation, makes them suitable tools for classification and data analysis.

## 1.2 Statement of the problem

Typically, fuzzy rule-based systems are generated using two ways: expert-driven approach and data-driven approach. In the first approach, the values of fuzzy rule parameters such as interval boundaries and membership functions are defined and set manually by an expert in the problem under study; while, these values, in the other approach, are defined automatically from a set of representative examples using a learning method (Kaufmann, Meier, & Stoffel, 2015).

Recently, data-driven rule generation methods have dominated the development of fuzzy rule-based systems for various reasons including the efficiency and low cost of the systems developed in addition to the availability of data sets.

In contrast to expert-driven approach, which maintains the interpretability of fuzzy rule-based systems, data-driven approach's interpretability is usually lost during the training process (J. Casillas, Cordón, Herrera, & Magdalena, 2003a). As a result, fuzzy rule-based systems built by machine learning algorithms usually become kind of black-box systems whose prime concern is getting highly accurate classification models rather than being used also for interpretation and analysis (M. Antonelli, Ducange, & Marcelloni, 2014).

Actually, the ability of fuzzy rule-based systems to produce understandable fuzzy rules with linguistic terms is considered as the "unique selling point" of fuzzy rule-based systems (Nauck, 2003). But this feature can be only maintained in fuzzy rule-based systems if some constraints are considered during the learning phase of these systems (J. M. Alonso & Magdalena, 2011b).

In order to fulfill the two modeling objectives, namely, interpretability and accuracy, several studies proposed methods to consider both of them during the learning process of fuzzy rule-based systems (J. M. Alonso & Magdalena, 2011a; Fernández et al., 2015; Gacto, Alcalá, & Herrera, 2012; Ishibuchi & Nojima, 2007). As there is a trade-off between interpretability and accuracy because they represent conflicting objectives, evolutionary algorithms, due to their efficiency, have been widely used to maximize the trade-off by finding accurate as well as interpretable fuzzy rule-based systems (M. Antonelli et al., 2014; Fernández et al., 2015; F Herrera, 2008). One of the most efficient algorithms in the literature to address the problem of accuracy-interpretability trade-off in fuzzy rule-based systems was proposed by Ishibuchi and Nojima (2007). The authors used a well-known multi-objective genetic algorithm (MOGA) called NSGA-II (Deb, Pratap, Agarwal, & Meyarivan, 2002) to find non-dominated fuzzy rule-based systems. The authors suggested that the use of more efficient MOGA may enhance the search ability of their proposed method (Ishibuchi & Nojima, 2007). An enhanced version of NSGA-II called Controlled Elitism NSGA-II was proposed that allows for a better distribution of solutions and faster convergence comparing with original NSGA-II (Deb & Goel, 2001). So, the use of Controlled Elitism NSGA-II in (Ishibuchi & Nojima, 2007) instead of the original NSGA-II may produce fuzzy rule-based systems with better performance.

In addition, the number of generated rules in fuzzy rule-based systems tends to be very high in high-dimensional classification problems which leads to the lack of global understanding of the fuzzy system (Mencar & Fanelli, 2008). Thus, producing a moderate number of fuzzy rules with relatively few antecedent conditions for high-dimensional problems helps to improve the understandability of fuzzy rule-based systems and increase their use in such kind of problems. In their work, Ishibuchi and Nojima (2007) proposed an efficient method for generating an initial set of fuzzy rule-based systems and concluded that in high-dimensional classification problems the performance of their final solutions depends on the quality of the candidate rules. Thus, the improvement of their method used to generate the initial rules may improve further the performance of the generated fuzzy rule-based systems.

Furthermore, the classification output or class label in a fuzzy rule-based system is associated with a certainty degree that reflects the confidence degree of the classification. This feature is considered as a good property because the user, based on this information, can either accept or reject the classification especially in cases where the cost of misclassification is high. In addition, rejected classification in the Winner Rule Reasoning method, which is one of the most used reasoning methods to calculate the classification output, occurs when there are at least two rules among the fired rules that have the same maximum value of compatibility grade with a given pattern but with different classes. Furthermore, a pattern which is not covered by any rule means that the compatibility grades of all the rules with this pattern are equal to zero. In this case, and since no class label would be provided by the classifier, this pattern is considered as wrongly classified. In rejected and uncovered classification cases, we can handle the classification either manually or using more reliable methods (Giorgio Fumera, Roli, & Giacinto, 2000).

Ensemble methods, which are known for their high classification accuracy, can be used as a reliable classifier to support fuzzy rule-based systems in uncertainty, rejected and uncovered pattern classifications and this may help decrease the error rates of fuzzy rule-based systems.

## 1.3 Research questions

The following research questions are used as guidance to conduct this research at various stages and to achieve the research objectives:

Q1- How can we improve the interpretability-accuracy trade-off in fuzzy rule-based systems?

Q2- How can we select a suitable fuzzy rule-based system from non-dominated fuzzy rule-based systems?

Q3- Has the quality of the fuzzy rule-based systems generated in the initial population of Genetic algorithms any effect on the performance of the final fuzzy rule-based systems?

Q4- which is more accurate: considering only one classifier or combine different base classifiers in one ensemble method?

Q5- Is it better to consider all the ensemble outputs or select a subset of them?

Q6- What is the most suitable method for building an accurate ensemble method?

Q7- How can the ensemble and fuzzy methods be integrated to improve the accuracy of a fuzzy rule-based system while maintaining its interpretability?

### 1.4 Research main aim and objectives

The main aim of this study is to propose an interpretable and accurate fuzzy-ensemble method that can be used for both classification and data analysis. This method is the result of combining the interpretability of fuzzy rule-based systems and the accuracy of ensemble methods.

In order to achieve this goal, the following intermediate objectives are identified:

1- Propose two variant methods of the work proposed in (Ishibuchi & Nojima, 2007) aiming to improve its ability to handle the problem of accuracy-interpretability trade-off in fuzzy rule-based systems. The objectives of the two variant methods, which are named as Proposal1 and Proposal2, are the following:

1.1 Proposal1 aims to improve the ability of the original algorithm to find non-dominated fuzzy rule-based systems with better interpretability-accuracy trade-off. This can be achieved by replacing the multi-objective genetic algorithm called NSGA-II used in (Ishibuchi & Nojima, 2007) by an enhanced version of NSGA-II called Controlled Elitism NSGA-II.

1.2 Proposal2 aims to improve the quality of the non-dominated fuzzy rule-based systems especially those extracted from high dimensional data sets by allowing the GA algorithm to start from a good initial population. In Proposal2, we used, in addition to Controlled Elitism NSGA-II, a feature selection-based method to improve the quality of the initial fuzzy rule-based systems generated in the initial population.

2- Analyze the performance of five different ensemble methods built using five different base classifiers.

3- Build an accurate ensemble method by proposing a design of an ensemble method that combines five different base classifiers and use a GA-based selection method to select a subset from all the ensemble outputs using accuracy and diversity measures as two objectives in the fitness function.

4- Propose a combination method that aims to improve the accuracy of the fuzzy rule-based system by using the accurate ensemble method to classify the patterns that have low certainty degree or in cases of rejected and uncovered classifications.

5- Evaluate two different methods for calculating the threshold value of the certainty degree under which the ensemble method is used for classification instead of the fuzzy rule-based system.

6- Evaluate the proposed fuzzy rule-based system and the designed ensemble method separately with their related works to assess their ability to be used as separated methods.

7- Evaluate the proposed fuzzy-ensemble method by comparing it with its constituents, namely, fuzzy rule-based method and the ensemble method.

## 1.5 Research scope

The current study addresses the problem of preserving the interpretability while optimizing the accuracy of fuzzy rule-based systems for classification problems. Specifically, we focus on combining the accuracy of the ensemble method with the interpretability of the fuzzy rule-based system to improve the accuracy of the latter while maintain its interpretability. For fuzzy systems, we study the methods used to maintain the interpretability and propose enhancements to some components of a well-known method. For this purpose, we propose two variant methods of the work in (Ishibuchi & Nojima, 2007) aiming to improve its ability to handle the problem of accuracy-interpretability trade-off in fuzzy rule-based systems. In addition, we investigated the performance of different ensemble methods in order to select the most accurate ensemble classifier that will be used for the fuzzy-ensemble method. To achieve this objective, we selected five different classifiers known for their classification ability and used them as base classifiers for five different ensemble methods. In addition, we explore the possibility of combining different base classifiers in one ensemble method. Furthermore, we test the effect of selecting a subset of the ensemble outputs rather than considering all of them on the performance of the ensemble method. After building an interpretable fuzzy rule-based system and an accurate ensemble method, we study the performance of the combined fuzzy-ensemble method using two criteria, namely, accuracy and interpretability. In addition, we study how we can use ensemble methods to enhance the accuracy of the fuzzy rule-based system by supporting its classification decisions in the uncertainty, rejected and uncovered cases.

We applied the proposed methods on six benchmark data sets taken from different classification problems and have different numbers of features, from 8 to 60, and different sizes, from 178 to 768 patterns. These data sets have been used by many researchers to evaluate their classification algorithms.

For error estimation, we run, for each data set, 10 independent iterations (with different data partitions) of 10-cross validation procedure (10×10cv). Then we calculate the average error rates over the 100 runs for each data. The results obtained are compared with other benchmark methods in terms of accuracy and interpretability.

## 1.6 Structure of the thesis

The remaining of the thesis is organized as follows.

**Chapter 2** presents an overview of fuzzy rule-based systems and different methods that have been employed to generate and optimize them. In addition, genetic-fuzzy systems are reviewed with special emphasis on the growing use of the multi-objective genetic algorithms to address the problem of finding a suitable balance between the interpretability and the accuracy in fuzzy rule-based systems.

**Chapter 3** introduces the concept of interpretability in fuzzy rule-based systems and different constraints proposed in the literature to preserve this property in fuzzy rule-based systems. In addition, a discussion of the problem of interpretability-accuracy trade-off is carried out and different approaches and techniques used to solve this problem are presented. Furthermore, a review of the use of the ensemble method concept in fuzzy systems and the approaches adopted to improve the quality of the classification is given. At the end of the chapter, a number of concluding remarks are summarized to highlight the most important points and issues raised in this chapter.

**Chapter 4** describes the proposed methodology which consists of three main phases: the first one aims to generate an interpretable and relatively accurate fuzzy system while the objective of the second phase is to construct an accurate ensemble method. The last phase combines the interpretability of fuzzy rule-based system with the accuracy of the ensemble method in one fuzzy-ensemble classification method.

**Chapter 5** discusses the results obtained from applying the proposed method on six benchmark data sets that represent different classification problems. The results are compared with existing methods and the discussion is carried out specifically in the context of interpretability-accuracy trade-off.

**Chapter 6** concludes this thesis with the main contributions and findings of this research and puts forth some proposals for future work.

**Appendix A** includes results of the non-dominated fuzzy rule-based systems generated from phase1.

**Appendix B** includes a table which lists the abbreviations of the ensemble methods used in Phase2 of the proposed method.

# CHAPTER 2: FUZZY RULE-BASED SYSTEMS AND GENETIC-FUZZY SYSTEMS

This chapter reviews some basic concepts of fuzzy set theory and fuzzy rule-based systems that will be used throughout this study. In addition, different methods employed to generate fuzzy rule-based systems are introduced from an interpretability point of view. These methods can be categorized into three categories: grid partition-based methods by which interpretable linguistic fuzzy rule-based systems can be extracted, and clustering-based methods which are characterized by their ability to induce highly accurate but less interpretable fuzzy rule-based systems, and finally hybrid-based methods. This latter group includes neuro-fuzzy and genetic-fuzzy systems. Due to the growing importance played by genetic algorithms, a special attention is given to hybrid genetic-fuzzy systems and learning tasks performed by these systems to define different components of fuzzy rule-based systems. In addition, the recent successful employments of multi-objective evolutionary algorithms (MOEAs) in finding good trade-offs between the accuracy and interpretability of fuzzy rule-based systems are also highlighted.

## 2.1 Fuzzy rule-based systems

### 2.1.1 Classical and fuzzy set

Fuzzy sets theory was introduced by (Zadeh, 1965) to extend the classical notion of sets. In classical set theory, an element either belongs or does not belong to a set. In fuzzy sets, however, the belonging of an element to a set is a matter of degree and its degree of belonging or membership is valued in the real interval [0, 1] by a *membership function*.

Formally, we can express the difference between the classical and fuzzy sets as follows.

Let $X$ be the universe of discourse, $A$ is a subset of the universe $X$, $x$ is an element where $x \in X$.

**a) - Classical sets**

In a classical set, the membership of $x$ in a classical subset $A$ is often viewed as a characteristic function, $\mu_A$ from $X$ to {0, 1} such that (Dubois & Prade, 1980):

$$\mu_A(x) = \begin{cases} 1, & iff^1\ x \in A \\ 0, & iff\ x \notin A \end{cases} \tag{2.1}$$

{0, 1}TA is called a *valuation set*.

**b)- Fuzzy sets**

As it is stated before, the valuation set in a fuzzy set is allowed to be in the real interval [0, 1] and the membership function $\mu_A$ can be written as follows:

$$\mu_A: X \rightarrow [0, 1] \tag{2.2}$$

Where $\mu_A(x)$ is the grade of the membership function of $x$ in $A$; where the closer the value of $\mu_A(x)$ is to 1, the more $x$ belongs to $A$.

$A$ is characterized by the set of pairs

$$A = \{(x, \mu_A(x)), x \in X\} \tag{2.3}$$

**c)- Some basic characteristics of fuzzy sets**

- ***$\alpha$-cut***

$\alpha$-cut $A_\alpha$ of $A$ is a set of elements in $X$ whose membership degrees are greater than a threshold $\alpha \in ]0, 1]$. It can be written as:

$$A_\alpha = \{x \in X,\ \mu_A(x) \geq \alpha\} \tag{2.4}$$

- ***Support***

The support of a fuzzy set $A$ is the ordinary subset of $X$:

$$supp\ (A) = \{x \in X, \mu_A > 0\} \tag{2.5}$$

- ***Core***

---

[1] N.B.: "iff" is short for "if and only if"

The core of a fuzzy set $A$ is the ordinary subset of $X$:

$Core(A) = \{x \in X, \mu_A(x) = 1\}$ (2.6)

- ***Height***

Height $hgt(A)$ of $A$ is the least upper bond of $\mu_A(x)$ and is written as:

$hgt(A) = sup_{x \in X} \mu_A(x)$ (2.7)

Figure 2.1 illustrates the concepts of *support*, *core* and *height* of a fuzzy set $A$.

- ***Normalized fuzzy sets***

$A$ is said to be ***normalized*** $iff \, \exists \, x \in X, \mu_A(x) = 1$ (2.8)

In this case, $hgt(A) = 1$.

- ***Convex fuzzy sets***

A fuzzy set $A$ is convex (figure 2.2) iff its $\alpha$-cuts are convex. Or can be defined equivalently as the following: $A$ is convex iff (Zadeh, 1965):

$$\forall \, x_1 \in X, \forall \, x_2 \in X, \forall \, \lambda \in [0, 1], \mu_A(\lambda x_1 + (1 - \lambda) x_2) \geq \min(\mu_A(x_1), \mu_A(x_2)) \quad (2.9)$$

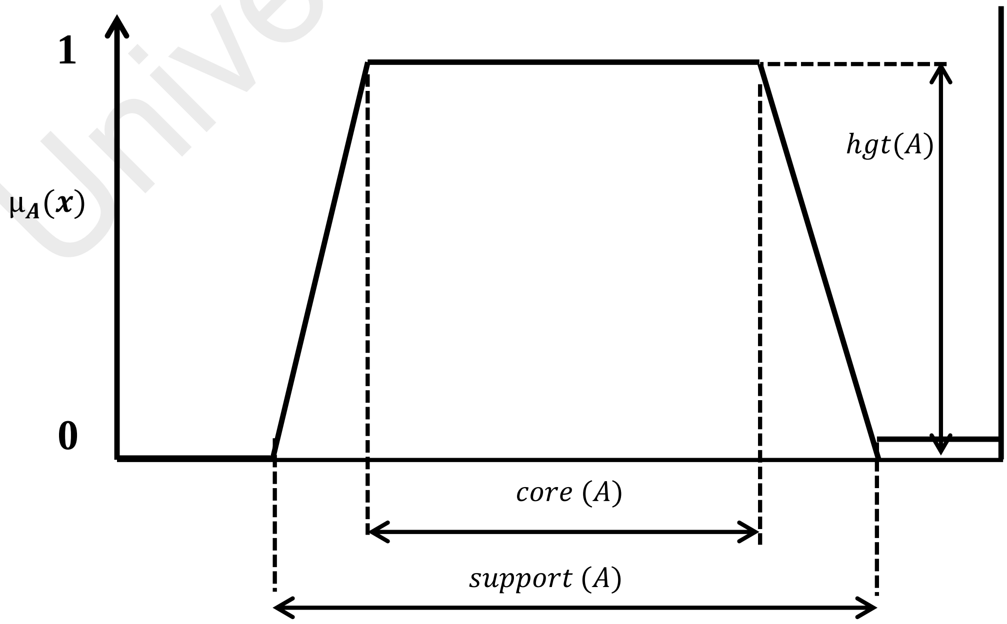


Figure 2.1 support, core and height of fuzzy set A

### 2.1.2 Basic concepts and issues of fuzzy rule-based systems

**a)- Components of fuzzy rule-based systems**

Fuzzy rule-based systems are computing framework based on the concept of fuzzy set theory, fuzzy if-then rules and fuzzy reasoning (J.S.R. Jang, Sun, & Mizutani, 1997). They are successfully applied in a wide variety of applications, such as data classification, decision analysis, automatic control, and pattern recognition. Fuzzy rule-based systems, because of their multi-disciplinary nature, are also known by other names like fuzzy logic controller (Lee, 1990), fuzzy expert system (Kandel, 1992), fuzzy model (Sugeno & Kang, 1988).

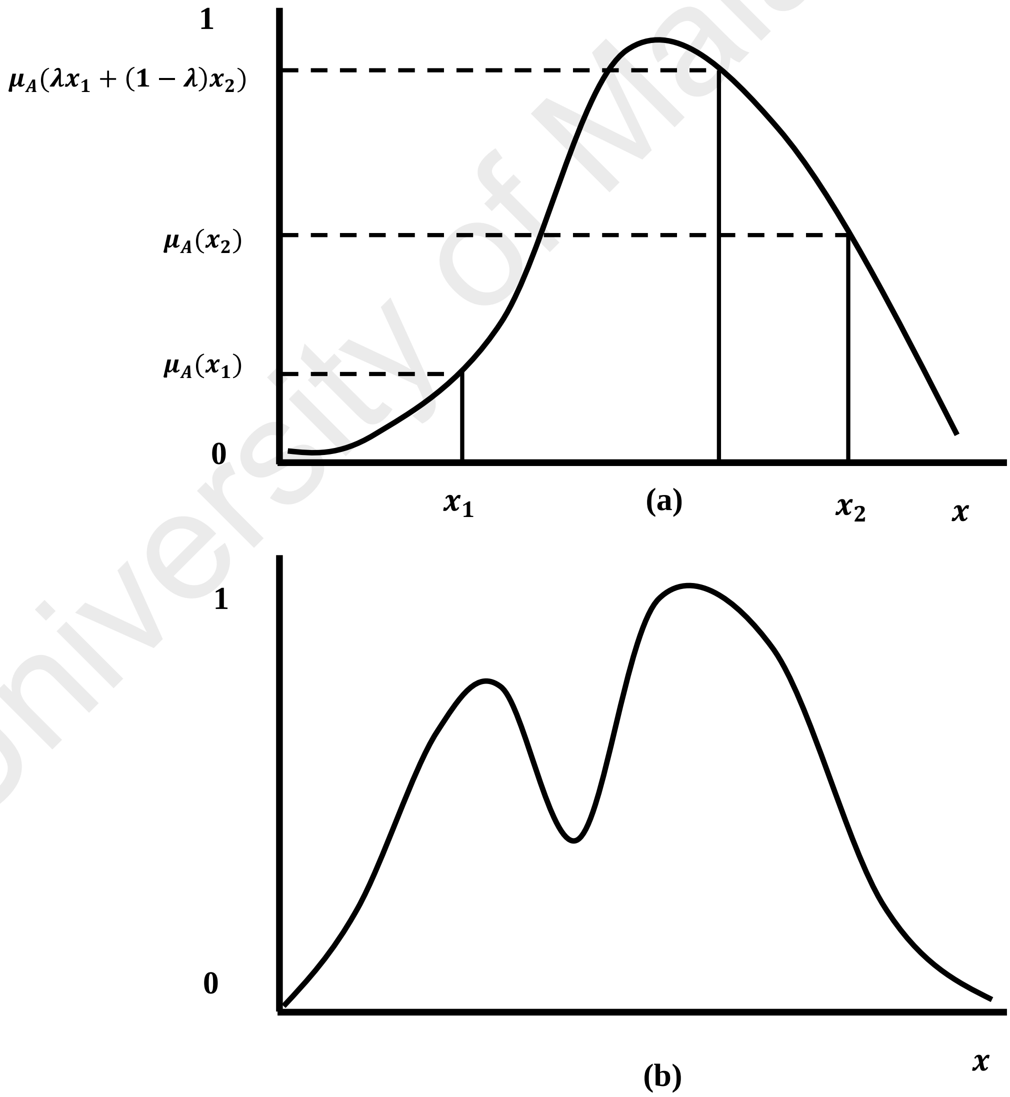


Figure 2.2 (a) Convex fuzzy (b) Non-convex fuzzy set (Dubois & Prade, 1980)

Basically, they consist of three main conceptual components: a *rule base*, which includes a set of fuzzy rules; a *database*, which defines the values of membership functions parameters used in the fuzzy rules; and a *reasoning mechanism*, which performs the inference procedure upon the rules to derive an output or conclusion (J.S.R. Jang et al., 1997).

Two other procedures are usually involved in the inference process, namely, *fuzzification* which converts the crisp inputs into fuzzy sets and *defuzzification*, which converts the fuzzy output into crisp value. Figure 2.3 shows the main components of a fuzzy rule-based system (J.S.R. Jang et al., 1997).

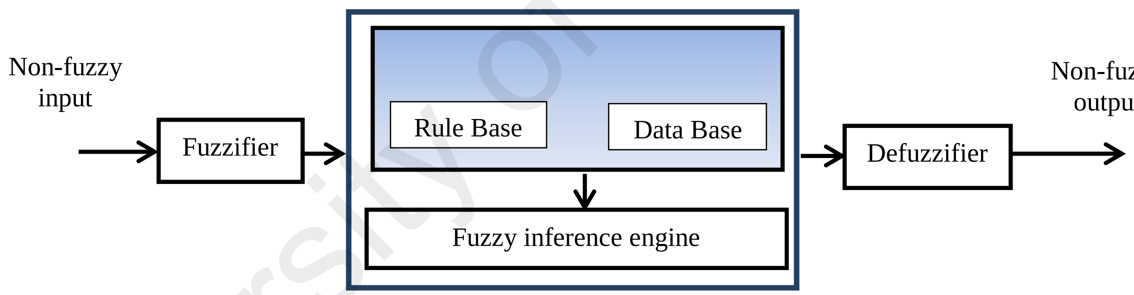


Figure 2.3 Block diagram for a fuzzy rule-based system

**b)- Fuzzy rule-based system modeling**

The construct of a fuzzy rule-based system, a process usually called *fuzzy modeling*, can be generally done in two ways:

- ***Expert-driven method***: the rule base are set manually based on prior knowledge that originates from *experts*, who are asked to express their knowledge in the form of If-then rules. The constructed systems are usually known as *expert-fuzzy system* (J.S.R. Jang et al., 1997).

- ***Data-driven method:*** the rules are generated from representative *data* on the problem under study using learning methods such as artificial neural networks (ANNs) or genetic algorithms (GAs). This process is generally called *fuzzy identification* (J.S.R. Jang et al., 1997).

The steps involved in the fuzzy rule-based system design are not usually the same and they are quite application-dependent but can be generally included in the following tasks (Yager & Filev, 1994):

1. A selection of the relevant input and output variables, either through the help of an expert of the domain under study or by applying a proper feature selection algorithm.

2. Choosing the appropriate type of fuzzy rule-based system, for example, either Mamdani or Sugeno fuzzy system. The selection of the model is usually guided by the modeling objectives, i.e. accuracy and interpretability. If, for example, the only objective is to produce an accurate fuzzy rule-based system, then it is better to select Sugeno fuzzy system; but if the interpretability is also important, then it is better to consider Mamdani fuzzy system.

3. Determination of the linguistic labels associated with each input and output variables. Definition of linguistic labels is usually done by an expert or by using some methods for fuzzy partitions.

4. Formation of the linguistic rules that describe the relation between the input and output variables; this phase is known as the fuzzy rule induction or learning. (This phase is explained in subsection 2.1.3)

5. Evaluation of the fuzzy rule-based system adequacy (evaluation phase).

In general, the resulting fuzzy rule-based system needs a further optimization process if the system's performance does not meet the specified targets. In this case, the values of the system's parameters are adjusted to the optimum values using optimization techniques. Choosing the appropriate parameters of a fuzzy rule-based system such as the family of membership functions, fuzzy system type, inference operators are dependent on some aspects related to the application like the purpose of modeling and the available knowledge.

**c)- Fuzzy IF-THEN rules**

A fuzzy rule-based system consists of a set of rules in the form of IF/THEN statements where the IF part is called *antecedent* and the THEN part is called *consequent*. Formally, a fuzzy rule can be written as follow:

$$R_k: If\ x_1\ is\ A_1^k\ and\ \dots and\ x_n\ is\ A_n^k\ \ Then\ y_1\ is\ B_1^k\ and \dots and\ y_m\ is\ B_m^k \quad (2.10)$$

Where $R_k$ is the label of the *k-th* rule, $x_i$ and $y_j$ are the $i^{th}$ input and $j^{th}$ output variables of the fuzzy system, respectively. $A_i^k$ and $B_j^k$ are the fuzzy sets of $i^{th}$ input and the $j^{th}$ output, respectively.

There are two main models of fuzzy rule-based systems, namely, Mamdani (Mamdani & Assilian, 1975) and Takagi-Sugeno-Kang (T. Takagi & Sugeno, 1985) fuzzy systems. For Mamdani-fuzzy model, both $A_i^k$ and $B_j^k$ are fuzzy sets while only $A_i^k$ is fuzzy set while $B_j^k$ is either linear function or constant for Takagi-Sugeno-Kang model.

Fuzzy sets can be conveniently used to represent linguistic labels or terms such as *small*, *large*, etc. Usually, for each variable considered; a set of corresponding fuzzy sets is specified that covers the whole domain of that variable. The definition of fuzzy sets for a given variable, which is called *linguistic variable*, is known as the *fuzzy partition*. These fuzzy sets are usually defined by membership functions like triangular, trapezoidal and Gaussian function. Figure 2.4 shows an example of the fuzzy partition of the linguistic variable *age*. Each of the three linguistic labels, namely, young, middle-aged and old are represented by fuzzy sets with values reflecting their meaning in real-life for the problem under study. So there is a semantic association between the linguistic terms and their corresponding fuzzy sets.

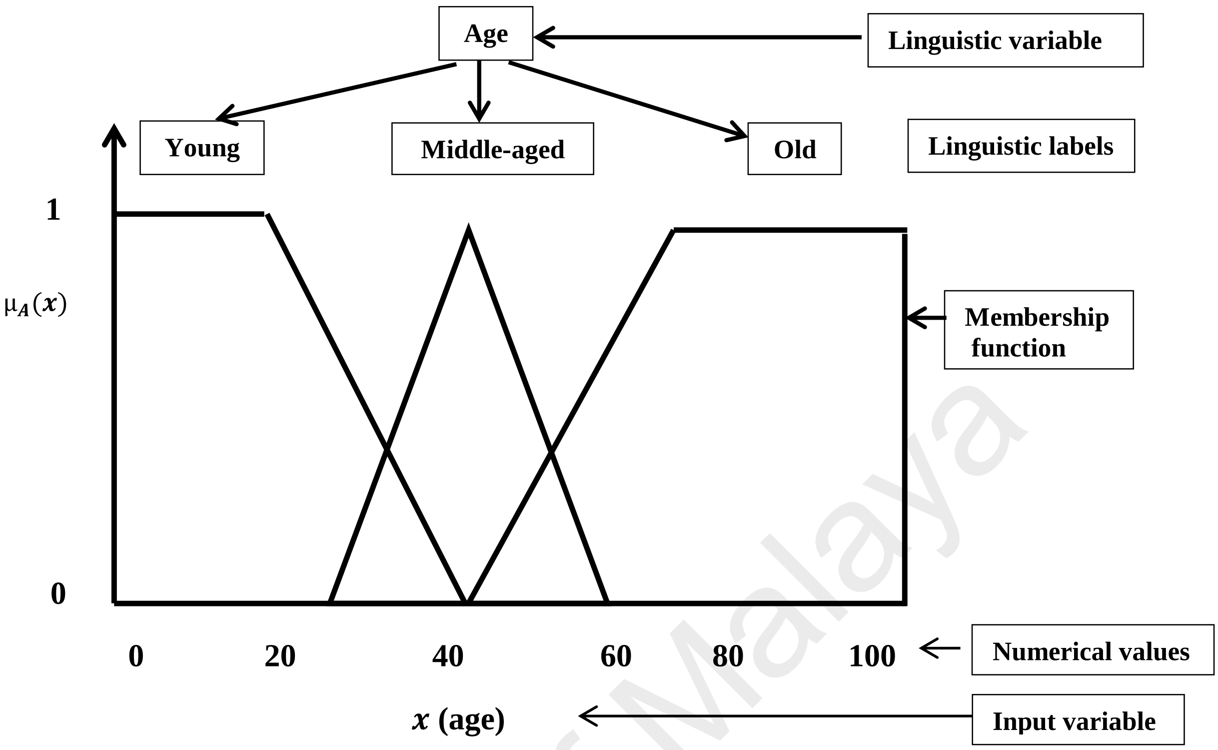

Figure 2.4 Fuzzy partition of input variable “age”



**d)- Fuzzy IF-THEN rules for classification problems**

For pattern classification problems and by using linguistic input variables, (2.10) can be written as (Oscar Cordón, del Jesus, & Herrera, 1999):

$$R_k\text{: } If\ x_1\ is\ A_1^k\ and\ x_2\ is\ A_2^k\ \ldots and\ x_n\ is\ A_n^k\ \ Then\ Class\ Y\ is\ C_j \tag{2.11}$$

Where $x_1, \ldots, x_n$ are linguistic input variables, $A_1^k, \ldots, A_n^k$ are linguistic labels representing the values of the linguistic input variables. Every input variable is divided into a finite set of linguistic labels such low, medium and high. $Y$ is the class $C_j$ in which the pattern belongs, where $j = \{1, \ldots, M\}$. So a given pattern $x_p = (x_{p1}, \ldots, x_{pn})$ is assigned to one of the M classes.

- ***Rules with certainty grade***

Another format which is commonly used in the literature related to the classification problems as in (Ishibuchi & Nojima, 2007) can be written as:

$$R_k: If\ x_1\ is\ A_1^k\ and\ x_2\ is\ A_2^k\ \dots and\ x_n\ is\ A_n^k\ \ Then\ Class\ Y\ is\ C_j\ with\ r^k \qquad (2.12)$$

Where $r^k$ is the certainty grade of the classification in the class $C_j$ .

A heuristic method (Ishibuchi, Nozaki, & Tanaka, 1992) can be used to determine the consequent class $C_j$ and the certainty grade $r^k$ of the rule $R_k$ in (2.12) as follows:

1- Calculate the compatibility grade $\mu_k(x_p)$ of each training pattern $x_p = \left(x_{p1}, \dots, x_{pn}\right)$ with the fuzzy rule $R_k$:

$$\mu_{R_k}\left(x_p\right) = \mu_1^k\left(x_{p1}\right) \times \dots \times \mu_n^k\left(x_{pn}\right) \qquad (2.13)$$

2- Calculate for each class, the sum of the compatibility grades for the training patterns with the fuzzy rule $R_k$:

$$\beta_{Class\ h}(R_k) = \sum_{xp\ \in Class\ h} \mu_{R_k}\left(x_p\right), \qquad h = 1, \dots, M \qquad (2.14)$$

3- Find class $C_j$ that has the maximum value of $\beta_{Class\ h}(R_k)$:

$$\beta_{Class\ C_j} = \max\{\beta_{class\ 1}(R_k), \dots, \beta_{class\ M}(R_k)\} \qquad (2.15)$$

We do not generate the fuzzy rule $R_k$ in the case where the consequent class $C_j$ of fuzzy rule $R_k$ cannot be uniquely determined.

4- Calculate the certainty grade $r^k$ as follows:

$$r^k = \{\beta_{Class\ C_j} - \bar{\beta} \Big/ \sum_{h=1}^{M} \beta_{Class\ h}\ (R_k), \qquad (2.16)$$

Where $$\bar{\beta} = \sum_{\substack{h=1 \\ h \neq C_j}}^{M} \beta_{Class\ h}\ (R_j)/(M-1) \qquad (2.17)$$

- ***Multiple consequent classes***

A third format of a fuzzy rule (Oscar Cordón et al., 1999) may include multiple consequent classes with their corresponding certainty grades and it can be written as:

$R_k: If\ x_1\ is\ A_1^k\ and\ \dots and\ x_n\ is\ A_n^k\ \ Then\ C_1\ with\ Cr_1^k\ and\ \dots and\ C_M\ with\ Cr_M^k$ (2.18)

where $Cr_j^k$ is the certainty grade of the rule $R_k$ to classify a pattern which belongs to the subspace of the rule into the class $C_j$. In this case, not only the class which has the maximum value in (2.15) is considered but every class $C_j$ is given its corresponding certainty grade $Cr_j^k$. This format was in (Ishibuchi & Yamamoto, 2005) to make a comparison between the three types of fuzzy rules defined in (2.11), (2.12) and (2.18).

The certainty grade $Cr_j^k$ for the class $C_j$ is different from the one in (2.16) and can be calculated by the following steps:

1- Calculate the compatibility grade $\mu_k(x_p)$ of each training pattern $x_p = (x_{p1}, \dots, x_{pn})$ with the fuzzy rule $R_k$as follows:

$$\mu_{R_k}(x_p) = \mu_1^k(x_{p1}) \times \dots \times \mu_n^k(x_{pn}) \qquad (2.13)$$

2- For each class, calculate the sum of the compatibility grades for the training patterns with the fuzzy rule $R_k$:

$$\beta_{Class\ j}(R_k) = \sum_{xp\ \in Class\ j} \mu_{R_k}(x_p), \qquad j = 1, \dots, M \qquad (2.14)$$

3- Calculate the certainty grade $r_j^k$ for each class:

$$Cr_j^k = \beta_{Class\ j}(R_k) \Big/ \sum_{j=1}^{M} \beta_{Class\ j}\,(R_k), \qquad (2.19)$$

The first two steps are the same steps used in calculating certainty grade $r^k$ but the difference lies in the third step. The expression (2.19) is also used in (Ishibuchi &

Yamamoto, 2005) to calculate the *confidence grade* of fuzzy rules, a measure frequently used in data mining to evaluate the association rule.

- ***Fuzzy reasoning method***

Fuzzy reasoning is the process of inferring a conclusion, or a class label in the case of classification problem, from a set of rules and a pattern. In classification problems, the most commonly used method for inferring the class label is the single winner rule (Oscar Cordón et al., 1999; Ishibuchi, Nakashima, & Murata, 2001).

### 2.1.3 Fuzzy rule-based system generation

Fuzzy rules can be generated using three kinds of fuzzy rule learning methods. The first kind uses grid partition to divide the input and output space and then optimize the grid structure according to the data set, while the second kind of methods employs clustering methods to separate the data into homogeneous clusters and then associate a rule to each cluster (Serge Guillaume, 2001). The third kind of methods can be called *hybrid methods*; it includes essentially soft computing methods such as artificial neural networks (ANNs) and genetic algorithms (GAs) (Serge Guillaume, 2001). The learning capabilities of these two methods have been employed to define the parameters of fuzzy rule-based systems. Due to the growing role, and especially in the recent years, that have been played by GAs in building interpretable fuzzy rule-based systems, we prefer for the sake of clarity and organization and due to the number of cited works to present these works in the separated section **"2.2 .genetic fuzzy systems"** rather than the current subsection (**2.1.3 Fuzzy rule-based system generation**).

**a)- Grid partitioning**

A grid partitioning is usually done by dividing the universe of discourse of the input variables into a number of intervals that represent the fuzzy sets, which are interpreted as linguistic labels and shared by all rules (see figure 2.4). Selecting the number of fuzzy sets per variable can be guided by experts or methods that aim to find the optimum number using some criteria (Serge Guillaume, 2001).

Using the grid partitioning, several approaches have been introduced in the literature to extract the rules. The following is a presentation of the three commonly used methods to generate the rules.

- ***Heuristic method***

Ishibuchi, Nozaki, Tanaka, Hosaka, and Matsuda (1994) proposed a heuristic method to generate rules with linguistic labels that are produced from grid partition. The universe of discourse of each input variable $x_i$ is evenly divided into $K_i$ symmetric triangular fuzzy sets, where $K_i = \{2, 3, 4, 5\}$. Figure 2.5 shows the fuzzy partitions for each input variable $x_i$. Then, all the rules corresponding to the possible combinations of the inputs' labels are generated. The total number of rules generated from *n* input fuzzy system is equal to $K_1 \times K_2 \times \ldots \times K_n$ where $K_i$ is the number of linguistic labels for the *i*-input. Actually, the choice of the granularity level has an important impact on the accuracy and the number of generated rules. In addition, although this method is suitable for learning interpretable fuzzy rules as the antecedent conditions are associated with linguistic labels that are fixed during the learning process but the number of generated rules can be very high when a problem with large input variables is considered. To find the optimum number of linguistic labels for each input variable and reduce the number of rules, Ishibuchi, Nozaki, Yamamoto, and Tanaka (1995) proposed a genetic rule selection method to choose the most relevant rules from all the generated rules. This heuristic method is commonly used in the literature to produce interpretable and linguistic fuzzy rule-based systems but it is usually followed by optimization process to improve both the accuracy and interpretability (see for example: (Ishibuchi & Nojima, 2007; Ishibuchi, Yamamoto, & Nakashima, 2005; Mansoori, Zolghadri, & Katebi, 2008). The steps needed to generate the fuzzy rules using this method are described in the section (**d)- Fuzzy IF-THEN rules for classification problems**).

- ***WM method: One rule per data pair***

Wang and Mendel (WM) (L. X. Wang & Mendel, 1992) proposed a method to induce the rules where the number of rules is limited by the number of the training pairs and not by the level of the granularity of the fuzzy partition. This method becomes a benchmark method for fuzzy rule generation due to its simplicity and efficiency (L. X. Wang, 2003). For example, WM method is used in (Jorge Casillas, Cordon, del Jesus, & Herrera, 2005; O. Cordón, Herrera, Magdalena, & Villar, 2001) as a rule learning module to generate the initial rules followed by an optimization phase using genetic algorithms. The induction process of WM method involves the five following steps (Serge Guillaume, 2001; L. X. Wang & Mendel, 1992):

1) - All the input variables are divided into a user defined number of fuzzy sets.

2)- Generate the fuzzy rules. One generated fuzzy rule $R_k$ corresponds to one pattern *i* of the data set. This rule can be written as follows:

$$R_k\!: if\ x_1\ is\ A_1^k\ and\ x_2\ is\ A_2^k\ \dots and\ x_n\ is\ A_n^k\ then\ y\ is\ C^k \qquad (2.20)$$

Where $A_i^k$ are the fuzzy sets that have the highest compatibility grades with $x_i^k$ for each input $i$ from the pattern $k$ the and the fuzzy set $C^k$ is the one with highest matching value with the observed output $y_k$.

3) - A grade value, which represents the fire strength of the considered pair, is assigned to each rule. If there are two or more rules with the same premises, we keep only the one with the highest fire strength value.

4- The induced rules and the rules provided by an expert can be merged in one rule base system.

5- The output is computed using a fuzzification procedure.

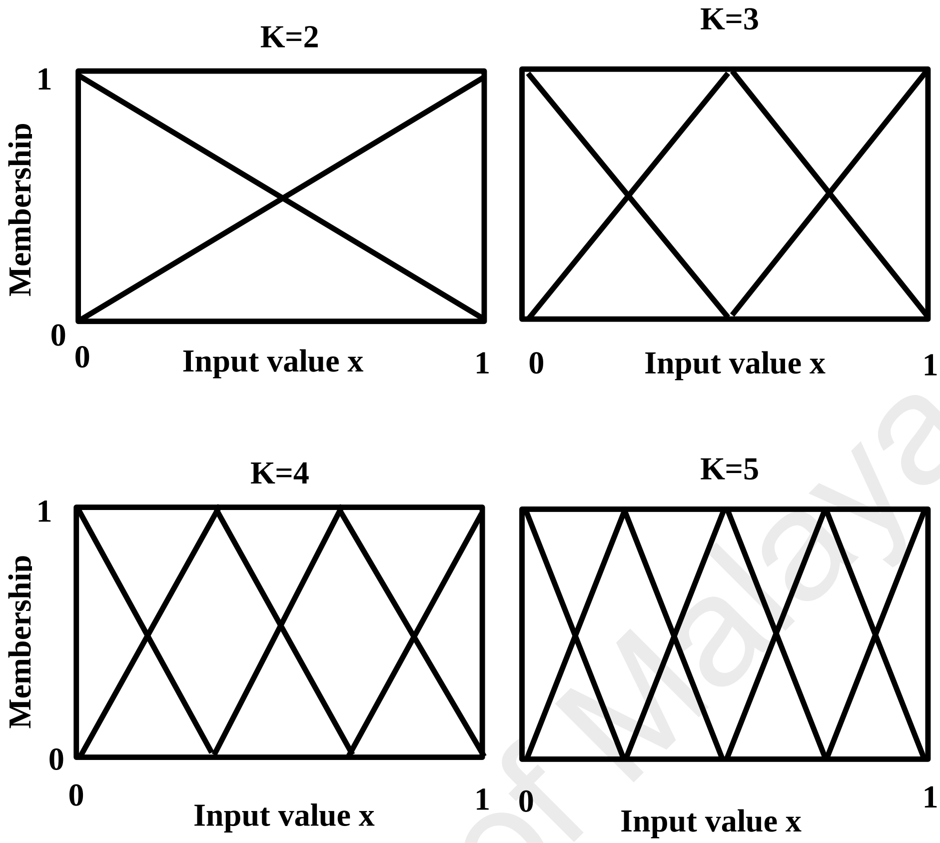


Figure 2.5 Fuzzy partitions of the input space (from (Ishibuchi, Nozaki, Tanaka, Hosaka, & Matsuda, 1993))

- ***Decision tree***

Decision trees (DTs) have been extensively used to solve classification problems in different domains. In fact, DTs have many desirable properties: they are interpretable classifiers, have the ability to handle numerical and categorical attributes and, have good performance especially when they are optimized or boosted by other methods like in ensemble method (Geurts, Ernst, & Wehenkel, 2006; Hullermeier & Vanderlooy, 2009).

The induction in a decision tree is an iterative process. Roughly speaking, the tree starts with one node (root) which represents the most meaningful variable (i.e. the one that maximize the information gain) and then generates a number of subnodes equal to the possible values of the selected variable. Then, a new node (input variable) is added for each subnode generated from the selected variable. This process is repeated until all leaves node (represent class labels) are reached.

From the interpretability point of view, the main advantage of decision trees is their ability to generate incomplete rules by choosing only the locally most relevant variables. This feature has been exploited by some researchers (José M Alonso, Magdalena, & Guillaume, 2008; Mikut, Jäkel, & Gröll, 2005; Pulkkinen, Hytonen, & Kolvisto, 2008; Pulkkinen & Koivisto, 2008) to perform a kind of feature selection method to reduce the dimension of input features by using some popular decision tree algorithms such C4.5 (J. R. Quinlan, 1993) and ID3 (J. Ross Quinlan, 1986). The generated decision tree is then converted into a fuzzy rule-based system which undergoes an optimization phase that aims to maximize its interpretability and minimize its complexity.

Fuzzy sets were also introduced in decision trees to create fuzzy trees usually through the modification of existing algorithms such as ID3 and C4.5 (Aymerich et al., 2011; Keeley Crockett, Bandar, McLean, & O'Shea, 2006; Cezary Z. Janikow, 1998; Olaru & Wehenkel, 2003; Witold Pedrycz & Sosnowski, 2000, 2001, 2005; Weber, 1992). The main advantage which fuzzy sets provide to fuzzy decision trees is to model the uncertainty around the split values of the attributes which results in soft instead of hard splits (Hullermeier & Vanderlooy, 2009). In this case, intervals in decision tree algorithms can be replaced with fuzzy sets and naturally with linguistic labels. In addition, fuzzy sets can offer more stability to decision tree methods which may result in improving the accuracy as is shown in (Keeley Crockett et al., 2006; Olaru & Wehenkel, 2003).

**b)- Clustering methods**

Unlike in the grid partition methods, fuzzy sets in clustering methods are not shared by all rules, but instead they are locally defined for each rule which makes their interpretation more difficult. Clustering methods are widely used and they proved their efficiency in building accurate fuzzy rule-based systems. These methods include *Fuzzy C-Means Clustering* introduced by (Dunn, 1973) and further improved by (Bezdek, 1981) and *Subtractive Clustering* method proposed by (Chiu, 1994). This latter was an improvement of a "*Mountain method* developed by (Yager & Filev, 1994).

**c) - Neuro-fuzzy systems**

Neuro-fuzzy systems are systems which employ the learning capabilities of neural networks to support the development of a fuzzy system (Nauck, Klawoon, & Kruse, 1997). The first neuro-fuzzy systems were developed in the domain of fuzzy control (Bastian, 1995; J. S. R. Jang, 1993; Shann & Fu, 1995; L. X. Wang & Mendel, 1992) but later they were applied in various domains such as data analysis (Nauck & Kruse, 1995), decision support system (Malhotra & Malhotra, 2002), and medical diagnosis (Nauck & Kruse, 1999b), financial problems (F Hoffmann, Baesens, Mues, Van Gestel, & Vanthienen, 2007), etc.

Hybrid Neuro-fuzzy systems are broadly divided into two categories (Sushmita Mitra & Hayashi, 2000):

- ***Fuzzy-neural system***: in which neural networks are equipped with the capability of processing fuzzy information (Keller & Hunt, 1985; S. Mitra & Kuncheva, 1995; Pal & Mitra, 1992). This is usually done by fuzzification of the ANNs components, the inputs and/or connection weights and/or outputs, so that the system can handle fuzzy information.
- ***Neuro-fuzzy system:*** in which the neural networks are employed in learning and adapting the fuzzy systems (Berenji & Khedkar, 1992; J. S. R. Jang, 1993; Keller & Tahani, 1992; Nauck & Kruse, 1997; H. Takagi, Suzuki, Koda, & Kojima, 1992). In this category, neuro-fuzzy systems are usually represented as a three-layer feedforward neural network where input layer represents the input variables, the

hidden layer presents the rules and the third layer represents the output variables while fuzzy sets are encoded as connection weights (Nauck & Kruse, 1997). To learn the parameters of the fuzzy system, the backpropagation-type learning algorithms is usually employed (Sushmita Mitra & Hayashi, 2000).

- ***Interpretability issue in neuro-fuzzy systems***

While the main concern of most of the proposed neuro-fuzzy systems such as GARIC (Berenji & Khedkar, 1992), ANFIS (J. S. R. Jang, 1993), FINEST (Tano, Oyama, & Arnould, 1996), FUN (Sulzberger, Tschichold-Gurman, & Vestli, 1993), EFuNN (Kasabov & Woodford, 1999), SONFIN (Juang & Lin, 1998) and FALCON (C. T. Lin & Lee, 1991) is to propose more accurate systems, some works consider the interpretability as an additional objective such as NEFCLASS (Nauck & Kruse, 1995), NEFCON (Nauck & Kruse, 1999a), (Paiva & Dourado, 2004), (Castellano, Castiello, Fanelli, & Mencar, 2005), (Vélez, Sánchez, Romero, & Andújar, 2010). In the latter kind of systems, neuro-fuzzy systems combine the interpretability of fuzzy systems with the learning capability of ANNs (Babuska & Verbruggen, 2003) to produce interpretable fuzzy rules from data set that can be understood by human being. In the following is a brief description of the main characteristics of NEFCLASS, one of the earliest and most popular interpretable neuro-fuzzy systems.

- ***NEFCLASS***

NEFCLASS which stands for **NE**uro-**F**uzzy for **CLASS**ification is a neuro-fuzzy system for classification problems. It is one the most discussed and used interpretable neuro-fuzzy systems in the literature. It has been used for extracting interpretable fuzzy rules that can be used for both classification and data analysis in different classification problems such as medical diagnosis (Nauck & Kruse, 1999b) and credit scoring problem (F. Hoffmann, Baesens, Martens, Put, & Vanthienen, 2002; F Hoffmann et al., 2007; Piramuthu, 1999).

As it is illustrated in Figure 2.6, NECLASS consists of three layers, the neurons of the first layer represent the input variables (two variables $x_1$ and $x_2$), while they represent the rules in the hidden layer (five rules, $R_1, \dots, R_5$). The neurons of the output layer represent the classes (two classes,$c_1$ and $c_2$). NEFCLASS uses fuzzy sets as weights between the input and the hidden layer, and the value 0 or 1 (represents the class) between the hidden and output layer.

The main feature of NEFCLASS is the use of share weights concept on some of the connections which makes sure that each linguistic label is represented by only one fuzzy set. For example in Figure 2.6, the fuzzy set $\mu_1^{(1)}$ represents the label "small" for the input variable $x_1$. The label "small" has the same interpretation for the both $R_1$ and $R_2$. In this case, during the learning process, the fuzzy set $\mu_1^{(1)}$ -which is shared by $R_1$ and $R_2$- is not allowed to change differently but it has to be tuned with the same manner in $R_1$ and $R_2$ and by this way the semantic of rule base will not be effected. In addition, the connections that share the same weights or fuzzy sets always come from the same input.

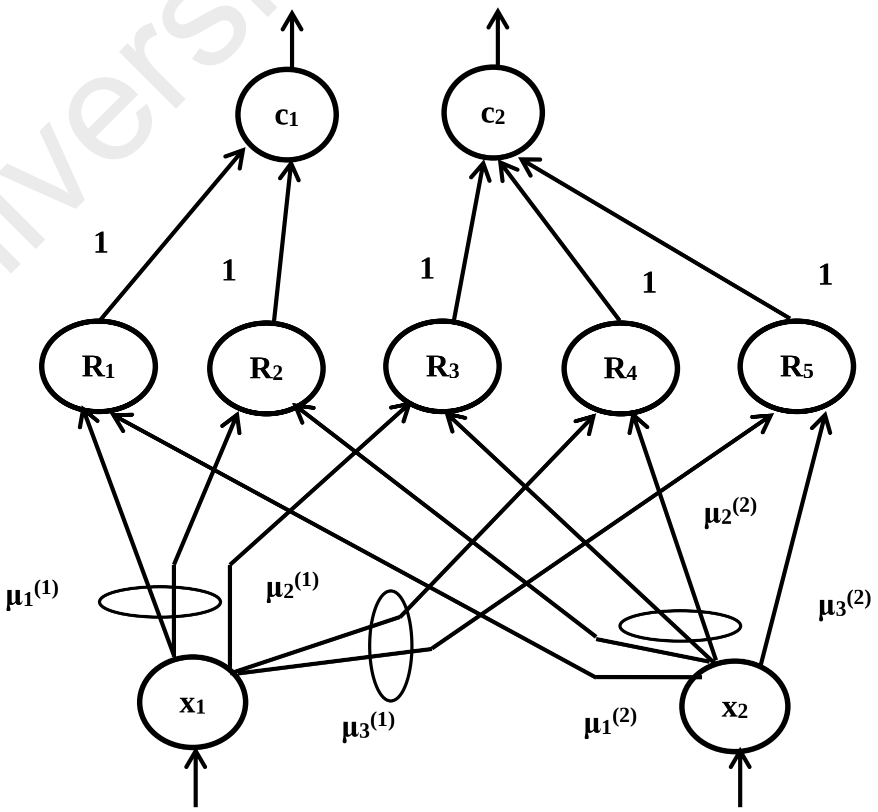


Figure 2.6 The structure of NEFLCASS system(Nauck & Kruse, 1995)

To generate a rule base from a data set, NEFCLASS performs three main tasks: (1) creation of an initial set of rules, (2) selection of the most relevant rules, (3) fine-tuning the fuzzy sets to increase the classification accuracy by employing a fuzzy heuristic variant of the gradient descent method known as fuzzy backpropagation, and (4) pruning the fuzzy rule base to reduce the complexity and enhance the readability (Nauck, 2001).

## 2.2 Genetic-fuzzy systems

### 2.2.1 Genetic algorithms

Genetic algorithms are computational models that use principles inspired by the evolutionary biology to find solutions to different optimization problems (Whitley, 1994). In these models, a *population* of candidate solutions, called *chromosomes*, to a given problem is evolved over successive iterations, called *generations*. For every generation, the chromosomes are evaluated with respect to the target problem using *fitness function* and the best solutions are given the chance to be selected using a *selection mechanism*. The selected chromosomes become parents and produced new chromosomes (offspring) through genetic operations such as *mutation* and *crossover*. Both selected and new chromosomes are combined to form the new generation which undergoes the same procedures (evaluation, selection and genetic operations). This process is repeated until the stopping criteria are met; for example, the population converges toward the optimal solution or the number of generations allowed is reached. Figure 2.7 is a flowchart that shows the way a simple genetic algorithm works.

GAs are powerful search algorithms which consider, simultaneously, many points in the search space and therefore they can avoid being trapped in local optima, and are highly adaptable for parallel computation (C. L. Karr & Gentry, 1993).

### 2.2.2 Genetic algorithms learning of Fuzzy rule-based systems

Genetic algorithms have been extensively used in the development of fuzzy rule-based systems. In the following, we are presenting an overview of the contributions that have utilized genetic algorithms for tuning and/or learning fuzzy rule-based systems

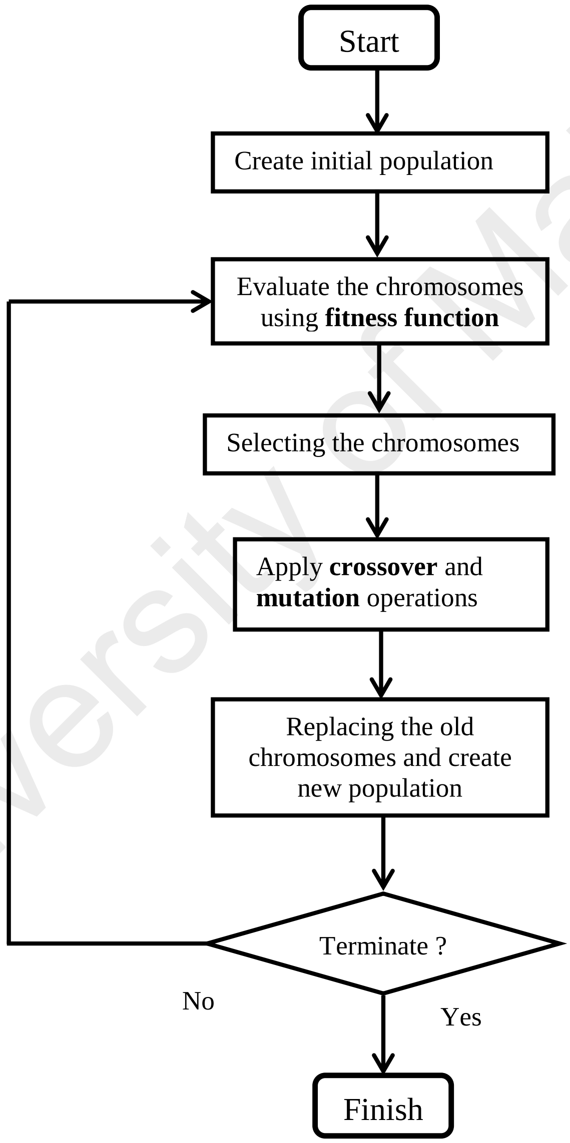


Figure 2.7 Flowchart of a genetic algorithm

**a)- Genetic tuning**

The objective of genetic tuning is usually to improve the performance of an existing fuzzy rule-based system. The tuning process is generally performed by adjusting the parameters of membership functions (MFs) which results in changing the shape of fuzzy sets without affecting the number of linguistic labels in each fuzzy partition (F Herrera, 2008). Tuning the membership functions has been widely adopted in fuzzy expert controller with a knowledge base built by an expert and the task of GAs is to enhance the performance of the system by adjusting the MFs values (see for example the following studies in Table 2.1 (Bonissone, Khedkar, & Chen, 1996; Glorennec, 1997; Hanebeck & Schmidt, 1996; C. Karr, 1991; C. L. Karr & Gentry, 1993). In other fuzzy systems which consider in addition to the accuracy, the semantic of rules, tuning the MFs is restricted within given intervals to preserve the original meaning of the knowledge base (Gürocak, 1999; Van Broekhoven, Adriaenssens, & De Baets, 2007). In addition, while maintaining the semantic of fuzzy rule-based systems, tuning of MFs can be boosted by introducing linguistic modifiers (Alcala, Alcala-Fdez, Gacto, & Herrera, 2007; Jorge Casillas et al., 2005) which provides more flexibility for membership tuning and expands the search space for GAs.

Beside the classic membership-based tuning method, other variants of tuning approaches have been suggested such as Genetic adaptive inference system (Alcalá-Fdez, Herrera, Márquez, & Peregrín, 2007; K Crockett, Bandar, & Mclean, 2007) and Genetic adaptive defuzzification method (Klösgen, 1996). In these two approaches, the methods which are used for inference and defuzzification processes are, unlike the common design practice, not a priori defined to certain fixed values but rather parameterized and the reason is to select, using GAs, among possible candidate methods for inference or defuzzification, the one that achieves the best performance.

Table 2.1 main characteristics of some proposed genetic fuzzy systems

| References | Coding scheme | MF type | Task of GA | Objective of using GA | Application and data sets |
|---|---|---|---|---|---|
| (C. Karr, 1991) | binary | Triangular | Tuning MFs | Accuracy | Controller |
| (C. L. Karr & Gentry, 1993) | binary | Trapezoidal | Tuning MFs | Accuracy | Controller: pH controller |
| (Homaifar & Mccormick, 1995) | integer | Triangular | define simultaneously the DB and RB (KB). | Accuracy | Controller : cart-centering problem |
| (Ishibuchi et al., 1995) | integer | Triangular | Rule selection | Accuracy/interpretability | Classification problem: iris data set. |
| (Hanebeck & Schmidt, 1996) | real | Gaussian | Tuning MFs. | Accuracy | Controller of Magnetic levitation system |
| (Bonissone et al., 1996) | Real | Trapezoidal | Tuning: the scaling factor and MFs. | Accuracy | Controller: the speed of freight train. |
| (Glorennec, 1997) | integer | Triangular | Tuning MFs | Accuracy | Controller: coordination between Autonomous robots. |
| (Cordon & Herrera, 1997) | Real, binary | Triangular | Rule selection, and tuning MFs. | Descriptive controller: accuracy/interpretability Approximate controller: accuracy | three three-dimensional control surfaces derived from mathematical functions |
| (Magdalena & Monasterio-Huelin, 1997) | Real, integer | trapezoidal | simultaneous define DB and RB (KB). | Accuracy | Control the synthesis of the biped walk of a simulated 2-D biped robot. |
| (Cordon, del Jesus, & Herrera, 1998) | real | Triangular | Rule and linguistic hedges selection, then tuning MFs. | Accuracy/interpretability | Classification problems: (1) Iris, and (2) Pima data sets. |
| (O Cordón, Del Jesus, Herrera, & Lozano, 1999) | Integer, real, angular. | Triangular | A selection of rules and tuning MFs. | Accuracy/interpretability | Regression problem: P1. |
| (Gürocak, 1999) | binary | Gaussian | Tuning MFs | Accuracy/interpretability | Controller: PD-like controller. |
| (O. Cordón et al., 2001) | Integer, real | Triangular | Define the DB parameters (granularity, the parameters of MFs and the working ranges) using GA and use WD method to generate the RB. | Accuracy | Regression problems: (1) P1 (2) P2 (3) P3 |

| References | Coding scheme | MF type | Task of GA | Objective of using GA | Application and data sets |
|---|---|---|---|---|---|
| (Jorge Casillas et al., 2005) | Real, integer, binary | Triangular | Tuning MFs parameters (basic parameters, non-linear scaling factor and the linguistic hedges) and selection method of the RB. MFs tuning and RB selection are performed in two modes: sequentially and simultaneously. | Accuracy/interpretability | (1) classification problem: rice taste evaluation problem (Nozaki, Ishibuchi, & Tanaka, 1997), (2) P2 |
| (Alcala, Alcala-Fdez, Gacto, et al., 2007) | real | Triangular | Lateral and amplitude tuning of MFs and rule selection. | Accuracy/interpretability | Regression problems: (1) P1 (2) P2 |
| (Alcalá-Fdez et al., 2007) | real | Triangular | Tuning the parameterized components of the adaptive fuzzy inference methods (conjunction, rule connective and defuzzification methods). | Accuracy/interpretability | (1) P1 (2) P2 (3) time series of sunspots (Foukal, 1990). |
| (Van Broekhoven et al., 2007) | Binary, real | trapezoidal | Tuning MFs. | Accuracy/interpretability | Classification problem: an ecological case study |
| (K Crockett et al., 2007) | real | Fuzzy regions around each decision node (fuzzy decision tree) | Tuning the parameterized fuzzy operators based on the T-norm Model. | Accuracy/interpretability | Classification problems: (1) Diabetes (2) Vehicle, (3) Mortgage, (4) Bank Loan. |

**P1**: the estimation of the low voltage network real length in villages, **P2**: estimation of the electrical medium voltage network maintenance cost in towns, **P3**: the modelling of tridimensional function (Cordon & Herrera, 1997), **DB**: Data Base, **RB**: Rule Base, **WD**: Wang and Mendel method (L. X. Wang & Mendel, 1992), **MFs**: **M**embership **F**unction**s**,

**b)- Genetic fuzzy-rule base learning**

Genetic algorithms have been employed and especially in the last two decades to learn or define the proper configuration of different components of fuzzy rule-based systems. The learning task may focus on one specific component or, consider more than one component simultaneously. In the following subsection, we cite some contributions that used GAs for fuzzy rule-based system learning.

- ***Learning the rule base***

Traditionally, genetic algorithms have been employed to derive the rule base of linguistic fuzzy rule-based systems from predefined linguistic labels. For such case, the universe of discourse of each input variable is divided into a number of linguistic labels and the GAs are used to select the most appropriate labels to form the antecedent conditions of the fuzzy rules (Thrift, 1991). In high dimensional problems where the number of rules grows exponentially with the number of the variables, GAs are utilized to decrease the complexity of the rule base, by reducing the number of fuzzy rules through rule selection procedure (Ishibuchi et al., 1995). In this case, only the relevant rules are selected. Later this method has been extensively employed as part of more advanced genetic-fuzzy systems either in multi-stage or joint evolutionary learning processes (O Cordón et al., 1999; Ishibuchi & Murata, 1996).

- ***Learning the knowledge base***

Some GA methods aim to learn the whole knowledge base to benefit from the positive synergy and interaction between the data base and the rule base. In this case, the whole knowledge base definition is encoded in each chromosome. This approach was successfully applied at the beginning to design fuzzy controller (Homaifar & Mccormick, 1995; Magdalena & Monasterio-Huelin, 1997). Later, more sophisticated methods have been proposed to improve the joint learning of data base and rule base in order to improve both the accuracy and the interpretability of fuzzy rule-based system.

For example, in (O. Cordón et al., 2001), the authors introduced the *embedded knowledge base learning* method where the components of data base (granularity, working ranges, membership functions shapes for each linguistic variable) are defined in a learning process that wraps a basic rule generation. So the data base is defined in the first step and then use, in the subsequent step, a simple method to generate the rules for the defined data base. A similar approach has also been adopted in (Cordon, Herrera, & Villar, 2001; Ishibuchi & Murata, 1996). This approach of learning has the possibility of generating a better definition for fuzzy rule-based system components but it is computationally expensive (F Herrera, 2008).

### 2.2.3 Approaches of rule learning and encoding in GA learning

In his review article, F Herrera (2008) has distinguished between four approaches of rule learning and encoding using genetic algorithms. These learning styles have been adopted by researchers working on fuzzy systems, from genetic-based machine learning area, to design and develop fuzzy rule-based systems using genetic algorithms.

**a)- Pittsburgh approach**

In this approach, genetic algorithms operate on chromosomes which are complete solutions; as a result, all the rules are encoded in one chromosome. The population composes of a set of fuzzy systems and the genetic operators are applied at the level of these systems. This style of encoding is capable of representing and solving complex problems but needs more computational resources since the entire rule base is encoded in each chromosome. This approach has been widely used to learn and optimize fuzzy rule-based systems (Akbarzadeh, Sadeghian, & dos Santos, 2008; Carse, Fogarty, & Munro, 1996; Feldman, 1993; F. Herrera, Lozano, & Verdegay, 1995; Homaifar & Mccormick, 1995; Ishigami, Fukuda, Shibata, & Arai, 1995; C. Z. Janikow, 1996; C. L. Karr & Gentry, 1993; Park, Kandel, & Langholz, 1994; Sanchez, Couso, & Corrales, 2001; Shimojima, Fukuda, & Hasegawa, 1995; Tsakonas, 2006).

**b)- Michigan approach**

The chromosomes in this style of learning are partial solutions where each rule is encoded in one chromosome and the entire population corresponds to one rule base. Thus, the population which composes of a set of individual rules is evolved using operators applied at the rule level (F Herrera, 2008). Some studies have applied this learning approach to build fuzzy rule-based systems (O Cordón et al., 1999; Cordon & Herrera, 1997; Furuhashi, Nakaoka, & Uchikawa, 1994; Gonzalez & Perez, 1999; Nakaoka, Furuhashi, & Uchikawa, 1994).

**c)- Iterative Rule Learning (IRL)**

It is another approach in which multiple runs are performed and the best chromosome is selected in every run to be included in the set of the best chromosomes that represents the global solution (Venturini, 1993). This learning approach has been adopted by some studies such in (O Cordón et al., 1999; Cordon & Herrera, 1997; Gonzalez & Perez, 1999; F. Herrera, Lozano, & Verdegay, 1998).

**d)- Genetic cooperative-competitive learning (GCCL)**

In this approach, the chromosomes compete and cooperate at the same time and the rule base can be encoded by the whole population or a subset of it. A number of researchers have applied this approach to build their fuzzy systems, see for example, (Berlanga, del Jesus, Gacto, & Herrera, 2006; Berlanga, Rivera, del Jesus, & Herrera, 2010; Chien, Lin, & Hong, 2002; Ishibuchi, Nakashima, & Murata, 1999; Juang, Lin, & Lin, 2000; Mucientes, Vidal, Bugarín, & Lama, 2009).

### 2.2.4 Multi-objective genetic algorithms

Multi-objective genetic algorithms are classes of genetic algorithms which are used to solve problems that have multiple and even conflicting objectives (Konak, Coit, & Smith, 2006). There are two approaches in multi-objective genetic algorithms optimization. The first is to combine the various objective functions into a single function in a linear fashion using weight factors. The drawback of this approach lies in the determination of the optimal weight values that characterize the user preferences. The second approach finds non-dominated Pareto optimal set of solutions for all optimal compromises between the conflicting objectives. It is a practical approach as the decision maker can find solutions with different trade-off levels (Konak et al., 2006).

The main advantage of MOEAs over the other multi-criteria algorithms is the ability of MOEAs to obtain, simultaneously and in a single run, non-dominated solutions, while non-MOEAs need multiple runs to find non-dominated solutions. EMO is one of the most active research areas in the evolutionary computation field (Ishibuchi, 2007).

In their first generation, MOEAs' main characteristic was the focus on finding good mechanisms to combine the selection of non-dominated[2] solutions with maintaining diversity (Gacto, Alcalá, & Herrera, 2009).

The most well-known algorithms of this generation are the following: multi-objective genetic algorithm (MOGA) (Fonseca & Fleming, 1993), NichedPareto Genetic Algorithm (NPGA) (Horn, Nafpliotis, & Goldberg, 1994), and Non-dominated Sorting Genetic Algorithm (NSGA) (Srinivas & Deb, 1994). In the second generation of MOEAs, the use of elitism becomes a standard practice to enhance the convergence of MOEAs (Deb et al., 2002). A number of algorithms have been proposed, of which NSGA-II algorithm (Deb et al., 2002), Pareto Archived Evolution Strategy (SPEA) (E. Zitzler & Thiele, 1999), SPEA2 (E. Zitzler, Laumanns, & Thiele, 2001), and Pareto Archived Evolution Strategy (PAES) (Knowles & Corne, 2000) are among the most widely used multi-objective genetic algorithms in the literature.

[2] Non-dominated solutions' concept is explained in Chapter 4.

### 2.2.5 Application of Multi-objective genetic algorithms in fuzzy rule-based systems learning

MOEAs have been extensively used in the context of interpretability to solve the problem of the interpretability-accuracy trade-off in fuzzy rule-based systems. This problem can be written as:

$$Maximize\ accuracy\ (FRBS), \quad Maximize\ interpretability\ (FRBS) \qquad (2.22)$$

In the literature, the interpretability has been considered at different levels of fuzzy rule-based system components. For example, to enhance the interpretability or readability of the rule base, a two-objective genetic algorithm was employed for rule selection (Ishibuchi, Murata, & Turksen, 1997). The main aim of this method is to maximize the interpretability (or reduce the complexity) and the accuracy of the fuzzy rule-based system by reducing the number of fuzzy rules and the number of misclassified training patterns, respectively. Rule selection was also considered, in addition to other criteria such as fuzzy sets selection, in the subsequent studies of prof. Ishibuchi and his co-researchers in (Hamada, Nojima, & Ishibuchi, 2009; Ishibuchi et al., 1997; Ishibuchi et al., 2001; Ishibuchi & Nojima, 2006; Ishibuchi & Yamamoto, 2004; Yusuke Nojima & Ishibuchi, 2009). Other components of fuzzy systems are learned or tuned using MOEA such as tuning of membership functions (Botta, Lazzerini, & Marcelloni, 2008; Botta, Lazzerini, Marcelloni, & Stefanescu, 2009; Munoz-Salinas, Aguirre, Cordon, & Garcia-Silvente, 2008; Pulkkinen & Koivisto, 2008) or jointly performing rule selection and membership function tuning (Alcala, Gacto, Herrera, & Alcala-Fdez, 2007; Gacto et al., 2012). In addition, there is an increasing trend, in the recent studies, of using MOEA to define the whole knowledge base (Alcala, Ducange, Herrera, Lazzerini, & Marcelloni, 2009; J. M. Alonso, Magdalena, & Cordon, 2010; Michela Antonelli, Ducange, Lazzerini, & Marcelloni, 2009a, 2009b, 2010; Cannone, Alonso, & Magdalena, 2011; Cococcioni, Ducange, Lazzerini, & Marcelloni, 2007; Oscar Cordón, Herrera, Del Jesus, & Villar, 2001; Ducange, Lazzerini, & Marcelloni, 2010b; Ishibuchi et al., 2001;

Ishibuchi & Nojima, 2007; Ishibuchi, Yamamoto, et al., 2005; Setzkorn & Paton, 2005; H. Wang, Kwong, Jin, Wei, & Man, 2005a). This trend may due, in part, to the recent development of more efficient multi-objective genetic algorithms such as NSGA-II, SPEA and SPEA2 which are widely applied in fuzzy systems to find a good trade-off between the interpretability and accuracy. In addition to their efficiency, these MOEAs offer a more interactive mechanism for choosing the solution, by providing a set of non-dominated fuzzy systems with different trade-offs between the interpretability and accuracy which gives the opportunity to the users to select the appropriate solution based on their preferences.

In early studies that applied MOEAs to optimize the interpretability and accuracy of fuzzy systems, a scalar fitness function with random weights was employed (Ishibuchi et al., 1997; Ishibuchi et al., 2001; Ishibuchi, Yamamoto, et al., 2005), then, more sophisticated MOEAs were used. NSGA-II is the commonly used MOEA (Alcala et al., 2009; J. M. Alonso et al., 2010; Botta et al., 2009; Ducange et al., 2010b; Gacto et al., 2009; Hamada et al., 2009; Ishibuchi & Nojima, 2006; Ishibuchi & Nojima, 2007; Munoz-Salinas et al., 2008; Yusuke Nojima & Ishibuchi, 2009; Pulkkinen & Koivisto, 2008; H. Wang et al., 2005a) but there are also some recently developed techniques which proved their efficiency and applied in fuzzy systems such as SPEA2 (Alcala, Gacto, et al., 2007; Gacto, Alcala, & Herrera, 2010; Gacto et al., 2009; Gacto et al., 2012; Munoz-Salinas et al., 2008; Setzkorn & Paton, 2005), (2+2)PAES (Alcala et al., 2009; Michela Antonelli et al., 2009a, 2009b; Cococcioni et al., 2007). Table 2.2 summarizes the main characteristics of the contributions that employ MOEAs in the learning process of fuzzy rule-based systems.

## 2.3 Summary

In this chapter, we reviewed the methods used for generating and optimizing the fuzzy rule-based systems. We specifically focused on the application of evolutionary techniques to address the accuracy-interpretability trade-off in fuzzy rule-based systems. The growing number of recently published works that employed multi-objective genetic algorithms to develop interpretable fuzzy rule-based systems reflects the success of this approach to find a good balance between the interpretability and accuracy. In addition, the introduction of efficient multi-objective genetic algorithms such as NSGA-II and SPEA2 has apparently encouraged some researchers to apply them to solve the problems of interpretability in fuzzy rule-based systems.

Table 2.2 the main characteristics of works that used MOEAs to learn different components of fuzzy rule-based systems

| **References** | **Name of MOEAs** | **Task of MOEA** | | | | | **Main contribution** | **Application and data sets** |
|---|---|---|---|---|---|---|---|---|
| | | FS | GL | FR-Sel | FS-Sel | MF-T | | |
| (Ishibuchi et al., 1997) | Scalar function | | ✓ | ✓ | | | The first multi-objective genetic algorithms applied to optimize the accuracy and interpretability of FRBSs. | Classification problem: (1) Iris data set |
| (Ishibuchi et al., 2001) | Scalar function | | ✓ | ✓ | ✓ | | Producing an interpretable fuzzy system for high-dimensional classification problems with three-objective genetic algorithms. | Classification problem: three benchmark data: (1) Iris, (2) wine, (3) glass |
| (Oscar Cordón et al., 2001) | Fonseca and Flemming's MOGA | ✓ | ✓ | ✓ | | | Proposing a MOEA-based process for jointly performing feature selection and FRBS's components learning. | Classification problem: Sonar data set |
| (Ishibuchi & Yamamoto, 2004) | MOG local search | | ✓ | ✓ | ✓ | | The use of two rule evaluation measures applied in data mining, namely, *confidence* and *support* as pre-screening criteria of candidate rules for fuzzy rule selection. | Classification problem: three benchmark data sets: (1) wine, (2) Iris, (3) Australian credit scoring |
| (Ishibuchi, Yamamoto, et al., 2005) | Scalar function | | ✓ | ✓ | ✓ | | The use of hybrid Michigan-Pittsburgh genetic algorithm for accuracy and interpretability optimization. | Classification problem: (3) benchmark data set: (1) Iris, (2) wine and (3) sonar. |
| (Setzkorn & Paton, 2005) | SPEA2 | | | ✓ | ✓ | | The use of MOEA to optimize both the classification and the complexity (interpretability). | Classification problem: 16 benchmark data sets: (1) Bcw, (2) Car, (3) Cmc, (4) Crx, (5) German, (6) Glass, (7) Image, (8) Kr-v-kp, (9) Mushroom, (10) Nursery, (11) Promoters, (12) Sonar, (13) Splice, (14)Vehicle, (15) Votes, (16) Waveform. |
| (H. Wang et al., 2005a) | NSGA-II | | ✓ | ✓ | ✓ | ✓ | Proposing an agent-based evolutionary approach to extract interpretable and accurate FRBS. | Three data sets were used:<br>(1) Nonlinear Plant With Two Inputs and One Output (L. Wang & Yen, 1999).<br>(2) Lorenz System (Yaochu Jin & Sendhoff, 2003) |

| References | Name of | Task of MOEA | | | | | Main contribution | Application and data sets |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | (3) Iris Data |
| (Ishibuchi & Nojima, 2006) | NSGA-II | | ✓ | ✓ | ✓ | | The use of MOEAs for constructing an ensemble of FRBSs with high diversity. | Classification problem: (6) benchmark data set: (1) Wisconsin breast cancer, (2) Diabetes, (3) Glass, (4) Cleveland heart disease, (5) Sonar, and (6) Wine. |
| (Alcala, Gacto, et al., 2007) | SPEA2, NSGA-II | | | ✓ | | ✓ | Propose $SPEA2_{acc}$, a modified version of SPEA2, where the algorithm focuses, in its search, on the Pareto zone that has high accurate FRBSs with the least number of rules. A comparison is made between NSGA-II and $SPEA2_{acc}$. | Regression real-world problem: (1) P2 |
| (Cococcioni et al., 2007) | (2+2)-PAES | | | ✓ | ✓ | | The use of MOEA to optimize both the accuracy and the complexity (interpretability). The proposed method is compared with SVD-QR(Yen & Wang, 1999) and other MOEAs, namely, SOGA, NSGA-II, MOGA, and PAES. | Regression problem: three benchmark data sets: (1) Box-Jenkins Gas Furnace (BJGF), (2) the Mackey-Glass time series (MG) and (3) the Lorenz Attractor (LA) datasets |
| (Ishibuchi & Nojima, 2007) | NSGA-II | | ✓ | ✓ | ✓ | | Analysis of the interpretability-accuracy trade-off. | Classification problem: (6) benchmark data set: (1) Wisconsin breast cancer, (2) Diabetes, (3) Glass, (4) Cleveland heart disease, (5) Sonar, and (6) Wine. |
| (Pulkkinen & Koivisto, 2008) | NSGA-II | | | ✓ | ✓ | ✓ | The use of C4.5 algorithm to create a decision tree and then convert it to FRBS which is used to initialize the first population of NSGA-II. | Classification problem: six benchmark data sets: (1) Wisconsin breast cancer, (2) Pima Indians diabetes, (3) Glass, (4) Cleveland heart disease, (5) Sonar, (6) Wine |
| (Pulkkinen & Koivisto, 2008) | NSGA-II | | | ✓ | ✓ | ✓ | Develop a reasoning mechanism model for of a bio-aerosol detector that has the ability to distinguish between safe and harmful aerosols using FRBS. The requirement of the FRBS, namely, accuracy and interpretability is achieved through the use of MOEA. | Real classification problem: bio-aerosol detector to distinguish between safe and harmful aerosols |

| **References** | **Name of** | **Task of MOEA** | | | | | **Main contribution** | **Application and data sets** |
|---|---|---|---|---|---|---|---|---|
| (Munoz-Salinas et al., 2008) | NSGA-II, SPEA, SPEA2 | | | | | ✓ | The use of MOEAs: NSGA-II, SPEA and SPEA2 for tuning the fuzzy membership functions of a fuzzy visual system for door detection for autonomous robots. | Real classification problem: door detection (Image processing). |
| (Alcala et al., 2009) | (2+2)-PAES+NSGA-II | | | | ✓ | ✓ | The use of (2+2)PAES to find a good trade-off between the accuracy and the interpretability in FRBSs. The proposed method is compared with NSGA-II and Single-Objective Evolutionary Model. | Regression problems: 9 benchmark data sets:<br>(1) P1 (2) P2, (3) Abalone, (4) Weather Izmir, (5) Weather Ankara, (6) Treasury, (7) Mortgage, (8) Computer Activity, (9) Pole Telecommunication. |
| (Michela Antonelli et al., 2009a) | (2+2)-PAES | | ✓ | ✓ | ✓ | | The use of MOEA-(2+2)-PAES to concurrently define the granularity of the partitions and the rule base. | Regression problem:<br>(1) P2<br>(2) A highly non-linear function that represents a concrete compressive strength prediction (UCI). |
| (Michela Antonelli et al., 2009b) | (2+2)-PAES | | ✓ | ✓ | ✓ | | The use of MOEA-(2+2)-PAES to concurrently define the granularity of the partitions and the rule base. | Regression problem: three benchmark data sets: (1) P2 (2) Weather Ankara, (3) Mortgage |
| (Botta et al., 2009) | NSGA-II | | | | | ✓ | Propose a new index that was employed to build an interpretable and accurate FRBS using NSGA-II. | Regression problem: (1) Parametric function, (2) truck data set. |
| (Hamada et al., 2009) | NSGA-II | | ✓ | ✓ | ✓ | | The use of fuzzy rules to examine the effectiveness of inter-vehicle (IVC) communication which used to avoid traffic congestion. | Real classification problem: evaluation of the effectiveness of IVC using four classes that represent different levels of IVC effectiveness. |
| (Yusuke Nojima & Ishibuchi, 2009) | NSGA-II | | ✓ | ✓ | ✓ | | The incorporation of user preference into multi-objective genetic fuzzy rule selection for pattern classification problems. | Classification problem: (1) benchmark data set: (1) Wisconsin breast cancer data. |
| (Gacto et al., 2009) | SPEA2, NSGA-II, two-version of NSGA-II, $SPEA2_{acc}$, extension of | | | ✓ | | ✓ | Performing an analysis on the application of six different MOEAs to obtain interpretable and still accurate FRBSs. | Regression problems: (2) data sets, (1) P2<br>(2) The Abalone dataset(UCI) is related to the task of predicting the age of abalone from physical measurements. |

| References | Name of | Task of MOEA | | | | | Main contribution | Application and data sets |
|---|---|---|---|---|---|---|---|---|
| | $SPEA2_{acc}$ | | | | | | | |
| (Gacto et al., 2010) | SPEA2 | | | ✓ | | ✓ | Proposing a semantic interpretability index for linguistic fuzzy models. In addition, a comparison between MOEAs and single-objective GAs is performed. | Regression problems: 9 benchmark data sets: (1) Plastic strength, (2) Quake, (3) P2, (4) Abalone, (5) Stock prices, (6) Weather Ankara, (7) Weather Izmir, (8) Mortgage and (9) Treasury. |
| (J. M. Alonso et al., 2010) | NSGA-II | | ✓ | ✓ | ✓ | | Embedding HILK, a fuzzy modelling methodology for designing interpretable FRBSs, into MOEA to perform feature selection procedure and fuzzy partition learning. | Classification problem: Glass data set |
| (Ducange et al., 2010b) | NSGA-II | | | ✓ | ✓ | | Propose a MOEA-based method to build FRBS for imbalanced and cost-sensitive data sets. | Classification problem: 13 data sets with imbalanced data sets: (1) ecoli-0-1-3-7_vs_2-6, (2) shuttle0vs4, (3) yeastB1vs7, (4) shuttle2vs4, (5) glass-0-1-6_vs_2, (6) glass-0-1-6_vs_5, (7) page-blocks-1-3_vs_4, (8) yeast-0-5-6-7-9_vs_4, (9) yeast-1-2-8-9_vs_7, (10) yeast-1-4-5-8_vs_7, (11) yeast-2_vs_4, (12) nodules (M Antonelli, Frosini, Lazzerini, & Marcelloni, 2006), (13) mammography (Woods et al., 1993). |
| (Gacto et al., 2012) | SPEA2 | | | ✓ | | ✓ | The use of SPEA2, with some modifications, to design a fuzzy controller for heating, ventilating and air conditioning systems (HVAC) with specific requirements. | Real application: a fuzzy controller for HVAC system. |
| (Ishibuchi & Nojima, 2013) | NSGA-II | | | ✓ | ✓ | | The use of a method called Repeated double cross-validation to select one suitable fuzzy solution among non-dominated fuzzy solutions generated using NSGA-II. | Classification problem: 17 data sets: (1) Appendicitis, (2) Australian, (3) Bands, (4) Bupa, (5) Cleveland, (6) Dermatology, (7) Glass, (8) Haberman, (9) Heart, (10) Mammographic, (11) Pima, (12) Saheart, (13) Sonar, (14) Vehicle, (15) Wdbc, (16) Wine, (17) Wisconsin. |

| References | Name of | Task of MOEA | | | | | Main contribution | Application and data sets |
|---|---|---|---|---|---|---|---|---|
| (M. Antonelli et al., 2014) | (2 + 2)M-PAES | | | ✓ | ✓ | | The authors used MOEA-based approach to learn concurrently the rule and data bases of fuzzy rule-based classifiers (FRBCs). | Classification problem: 24 data sets: (1) Haberman, (2) Hayes-roth, (3) Iris, (4) Mammographic, (5) Newthyroid, (6) Tae, (7) Bupa, (8) Appendicitis, (9) Pima, (10) Glass, (11) Saheart, (12) Wisconsin, (13) Cleveland, (14) Heart, (15) Wine, (16) Australian, (17) Vehicle, (18) Bands, (19) Hepatitis, (20) Pasture, (21) Wdbc, (22) Dermatology, (23) Ionosphere, (24) Sonar. |
| (Fazzolari, Alcalá, & Herrera, 2014) | SPEA2 | | ✓ | ✓ | | ✓ | This study presents a fuzzy discretization procedure for granularities and fuzzy partitions of fuzzy rule based systems. The objective of the study is to improve the complexity-accuracy trade-off of fuzzy models using MOEA. | Classification problem: 35 data sets: (1) Iris, (2) tae, (3) hepatitis, (4) wine, (5) automobile, (6) glass, (7) newthyroid, (8) heart, (9) Cleveland, (10) haberman, (11) ecoli, (12) bupa, (13) balance, (14) crx, (15) Australian, (16) Wisconsin, (17) pima, (18) vehicle, (19) german, (20) contraceptive, (21) titanic, (22) segment, (23) spambase, (24) banana, (25) phoneme, (26) page-blocks, (27) texture, (28) optdigits, (29) satimage, (30) thyroid, (31) ring, (32) twonorm, (33) coil2000, (34) penbased, (35) magic. |

**FS**: feature selection, **GL**: granularity learning, **FR-Sel**: fuzzy rule selection, **FS-Sel**: fuzzy set selection, **MF-T**: membership function tuning,
**P1**: the estimation of the low voltage network real length in villages, **P2**: estimation of the electrical medium voltage network maintenance cost in towns,
**P3**: the modelling of tridimensional function (Cordon & Herrera, 1997).

# CHAPTER 3: INTERPRETABILITY IN FUZZY RULE-BASED SYSTEMS AND ENSEMBLE METHODS

This chapter is devoted to introduce the interpretability concept in fuzzy rule-based systems as well as different constraints that have been proposed in the literature to preserve this property during the learning process of fuzzy rule-based systems. These constraints are classified, according to the type of task they perform in maintaining the interpretability, into two categories: (1) semantic-based constraints, which aim to ensure the use of linguistic terms with clear semantic meaning, and (2) complexity-based constraints which aim to reduce the complexity of the system by for example reducing the number of rules and features. In addition, the methods that have been used to evaluate the interpretability property in fuzzy rule-based systems are reviewed. Furthermore, the approaches employed to balance the interpretability-accuracy trade-off are presented. In another but a related issue, a discussion is carried out on the use of the ensemble method concept in fuzzy rule-based systems and the approaches adopted to improve the quality of their classification accuracy. Finally, this chapter concludes by presenting a number of concluding remarks that highlight the most important ideas and issues raised in this chapter.

## 3.1 Interpretability concept in fuzzy rule-based systems

The concept of interpretability appears in many domains under different names such as understandability, comprehensibility, transparency, intelligibility, readability, etc. While most of the researchers considered these terms as synonymous (Zhou & Gan, 2008), some others made distinction between them especially in the fuzzy modeling context (Mencar & Fanelli, 2008; Riid, 2002). For example, according to Riid's opinion (Riid, 2002), transparency and interpretability do not carry the same meaning when used to characterize the fuzzy rule-based systems. The author considered interpretability as a default property of fuzzy rule-based systems being established with linguistic rules and fuzzy sets associated with these rules while transparency is not a default property and it measures the reliability or validity of the linguistic interpretation of the system. This distinction was not supported by some authors (Zhou & Gan, 2008) as they think that both transparency and interpretability share the same connotations according to the two definitions in practice and have been used in parallel in fuzzy rule-based system modeling (J. M. Alonso, 2007; Jiménez, Gómez-Skarmeta, Roubos, & Babuška, 2001; Nauck, 2000; Roubos & Setnes, 2001). In fact, there is generally agreement between researchers on the role or functionality of the interpretability or transparency property in the fuzzy rule-based system modeling. So what is the main function of interpretability or transparency property in a system? In system modeling, *transparency* according to M. Brown and Harris (1994) is a property that allows the user to understand the influence of each system parameter on the system output. Similarly, *interpretability* is defined by (J. Casillas, Cordón, Herrera, & Magdalena, 2003b) as the capability to express the behavior of a real system in an understandable way. More specifically, *Interpretability* in a fuzzy modeling context means that human beings are able to understand the fuzzy rule-based system's behavior by inspecting the rule base (Mikut et al., 2005). Or according to Bodenhofer and Bauer (2003) means the possibility to estimate the system's behavior by reading and understanding the rule base only.

By inspecting the previously stated definitions of the two terms, we can deduce that the main feature of an interpretable or transparent system is the ability to describe its behavior or the relationship between input(s) and output(s) in a way that is understandable for the human being. So the *understandability* of the system's behavior is the main functional goal of an interpretable or transparent system. Therefore, the two terms, namely, transparency and interpretability are used interchangeably in the current study unless otherwise stated.

Review of interpretability concept in fuzzy rule-based systems literature shows that this concept is not characterized by a well-defined measure or metric that can be used for interpretability evaluation like for example the case of accuracy concept (J. M. Alonso & Magdalena, 2011b; Bodenhofer & Bauer, 2003; J. Casillas et al., 2003b; De Oliveira, 1999; Serge Guillaume, 2001; Mencar & Fanelli, 2008; Nauck, 2000; Riid, 2002; Zhou & Gan, 2008). The reason of that is the blurry definition of this concept which firstly is subjective, because it depends on the person who makes the evaluation and secondly it is application-oriented (Mencar & Fanelli, 2008). But many researchers agreed on a number of constraints that should be set to ensure the interpretability in a fuzzy rule-based system. In the following section, we will discuss these constraints and their main contributions to the interpretability property in the fuzzy rule-based system modeling.

## 3.2 Interpretability constraints for fuzzy rule-based system

One of the common ways for designing interpretable fuzzy rule-based systems is to impose a set of constraints (or formal properties) on the system components such as fuzzy sets and rule base during the learning process (Cordon, 2011). In the following subsection, we are going to introduce these constraints, categorized under two classes, namely, semantic and complexity-based constraints. A summary of the works that used these constraints appears in Table 3.1.

### 3.2.1 Semantic-based constraints

The main objective of imposing semantic constraints during the fuzzy rule-based system construction is to preserve the semantics associated with the membership functions. In other words, the fuzzy partitions of a given variable can be interpreted as linguistics labels such as: low, medium and high. In addition, semantic constraints include also some other properties related to the logic side of the rules such as the consistency of the rules (Gacto, Alcalá, & Herrera, 2011). Semantic-based constraints can be divided into the following three classes according to the components on which the constraints are applied: constraints for fuzzy sets, fuzzy partition and fuzzy rules.

**a) - Semantic-based constraints for fuzzy sets**

- ***Normality***

Normality for a fuzzy set means that there exists at least one element or data point in the universe of discourse with full membership, i.e., has a membership value equal to 1 (see Figure 3.1). This can be formally stated with the following expression:

$$\exists\, x \in X, \qquad \mu_A(x) = 1 \tag{3.1}$$

For interpretable fuzzy rule-based systems, the linguistic terms should have a clear semantic meaning; that is, one element of the universe of discourse should exhibit full matching with the linguistic term semantically represented by the fuzzy sets (De Oliveira, 1999).

Normality is a requirement that is implicitly assumed by the overwhelming majority of literature related to the interpretability with few exceptions that were explicitly cited normality as one of the requirements for fuzzy rule-based systems' interpretability (De Oliveira, 1999; Serge Guillaume, 2001; S. Guillaume & Charnomordic, 2011, 2012; Mencar & Fanelli, 2008; Roubos & Setnes, 2001; Setnes & Roubos, 2000; Zhou & Gan, 2008).

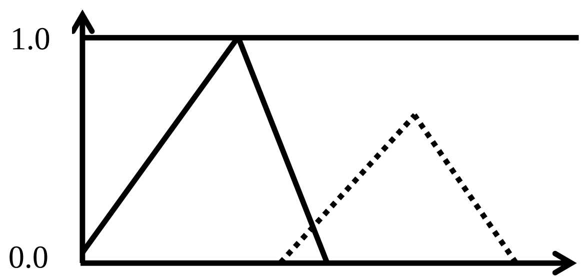


Figure 3.1 An example of non-normalized fuzzy set (fuzzy set with dotted points)

- ***Convexity***

A fuzzy set is a convex if the membership values of elements belonging to any interval are not lower than the membership values at the interval's extremes. Formally this can be written as follows:

$$\forall\, a, b, x \in X, \qquad a \le x \le b \;\rightarrow\; \mu_A(x) \ge \min\{\mu_A(a), \mu_A(b)\} \qquad (3.2)$$

It is semantically considered as a completion of the normality requirement. According to Mencar and Fanelli (2008), convexity assures that the concept represented by the fuzzy set is related to a single specific property of a perceived object. In other words, the concept can be conceived as elementary.

Convexity is very important requirement of interpretability and it is implicitly assumed in interpretable fuzzy rule-based systems except with few articles (De Oliveira, 1999; S. Guillaume & Charnomordic, 2012; Mencar & Fanelli, 2008; Zhou & Gan, 2008).

In fact, the normality and convexity of a fuzzy set can be easily satisfied by selecting the most commonly used membership function types such as triangular and Gaussian and this explains why most of the researchers do not explicitly discussed these semantic constraints in their works.

**b) - Semantic-based constraints for fuzzy partition**

- ***Coverage and completeness***

The universe of discourse of a variable is complete if every data point of element belongs at least of one of the generated membership functions. Formally, this can be written as follows:

$$\forall\, x \in X, \exists\, A \in F, \mu_A(x) > 0 \qquad (3.3)$$

Where $F$ is the set of fuzzy sets defined in the universe of discourse $X$.

The expression (3.3) suggests that for every data point, it is required that the membership value should not be zero for at least one of the fuzzy sets. That is, every data point is semantically represented by at least one of the linguistic terms (see Figure

3.2). This also means that a fuzzy rule-based system should be able to infer a proper conclusion for every input (De Oliveira, 1999). Completeness is justified by the fact that in human reasoning there will never be a gap of description within the range of the variable (Herrmann, 1997).

In interpretable fuzzy models, another definition of the completeness constraint is known as α-completeness can be defined with the following expression:

$$\forall x \in X, \exists A \in F, \mu_A(x) \geq \alpha \qquad (3.4)$$

α-completeness is preferred because it guaranties that every element in the universe of discourse is well presented by a fuzzy set with a minimum degree $\alpha$, given the rise to the concept of *strong coverage* (De Oliveira, 1999).

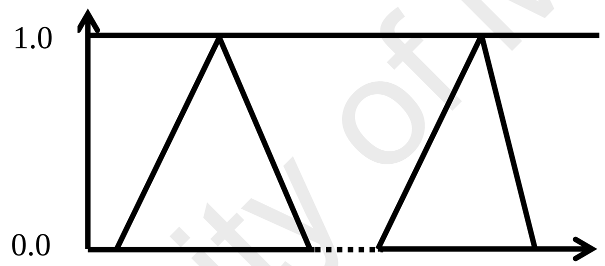


Figure 3.2 An example of bad coverage of the universe of discourse, some elements (potted points) in the universe of discourse are not covered

- ***Distinguishability***

Distinguishability means that each fuzzy set should be distinct enough from the other fuzzy sets defined on the same universe of discourse so they represent distinct concepts that can be assigned to linguistic terms with clear and different semantic meanings (De Oliveira, 1999; Mencar & Fanelli, 2008; Zhou & Gan, 2008).

Distinguishability is a basic and essential constraint that has been widely adopted in interpretable fuzzy modelling literature (see, for example, (De Oliveira, 1999; Espinosa & Vandewalle, 2000; S. Guillaume & Charnomordic, 2004; Mencar, Castellano, &

Fanelli, 2007; Mencar & Fanelli, 2008; Setnes, Babuska, Kaymak, & Van Nauta Lemke, 1998). This property offers a number of advantages for interpretable fuzzy modeling including: reduce redundancy, which may be present in the form of similar fuzzy sets that represent compatible concepts (Setnes et al., 1998), and more importantly the ease of the linguistic interpretation of the model since fuzzy sets represent well-separated concepts (Setnes et al., 1998). Thus, when the distinguishability is lost, especially during an accuracy-oriented learning process, it is difficult to assign distinct linguistic terms to fuzzy sets. An example of distinguishable and non-distinguishable fuzzy sets is shown in Figure 3.3. As can be seen in the Figure 3.3, it is easy to assign labels or linguistic terms such as: very low, low, average, large and very large to the distinguishable fuzzy sets while it is difficult to do that for non-distinguishable fuzzy sets as most of them represent almost the same concept. During the design process of fuzzy models, other constraints such as coverage may require overlapping fuzzy sets, and thus distinguishability should be carefully balanced with other constraints (Mencar & Fanelli, 2008).

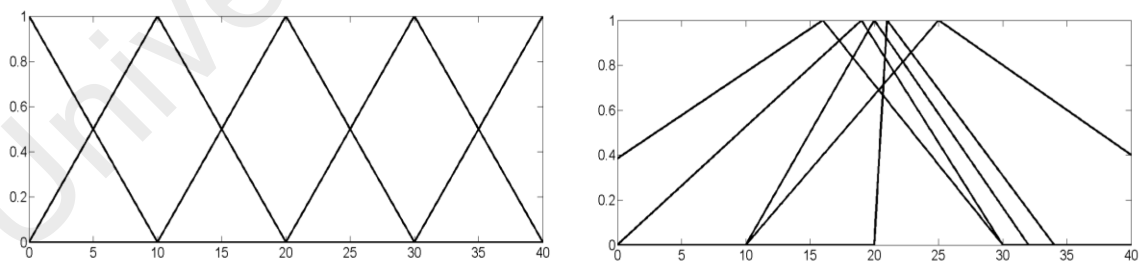


Figure 3.3 Example of non-distinguishable fuzzy sets (right) and distinguishable fuzzy sets (left)

**c) - Interpretability constraints for the rule base**

- ***Consistency***

Consistency means the absence of contradictory in the rule base, i.e., if two or more rules have similar antecedents, they should have different consequents (Dubois, Prade, & Ughetto, 1997; Serge Guillaume, 2001; Y Jin, von Seelen, & Sendhoff, 1999). In a knowledge base of expert systems, inconsistency in the rule base occurs when there are two rules in the form $A \rightarrow B$ and $A \rightarrow C$, where $B$ and $C$ are mutually exclusive concepts. Inconsistency means that a statement and its negation can be derived from the same knowledge base which makes it useless. In fuzzy logic, consistency is a matter of degree since the statement $A \wedge \neg A$ can be true with a certain degree greater than zero. Thus, in the fuzzy rule base, partial inconsistency can be tolerated if it is acceptably small (Mencar & Fanelli, 2008). Checking the fuzzy rule base for consistency remains an important constraint for its validation and interpretation (Dubois et al., 1997; Mencar & Fanelli, 2008).

- ***Type of fuzzy rules***

Actually, there are two well-known types of fuzzy rule-based systems, namely, Mamdani and Takagi systems. The only difference between these two models lies in their consequent part; as Mamdani model uses fuzzy set in its consequent part while Takagi model uses a linear real function. Actually, Mamdani model is more interpretable because a fuzzy set is suitable to express human perception knowledge while the linear function in the consequent of Takagi model does not represent any physical meaning (Zhou & Gan, 2008). For this reason, most of the researchers whose interpretability is their main objective have been using Mamdani model to build their fuzzy rule-based systems (Cordon, 2011).

### 3.2.2 Methods applied to achieve the semantic-based constraints

Much of the works related to achieve the semantic-based constraints are attributed to developing methods that aim to maintain the distinguishability property in fuzzy rule-based systems. This property can be characterized using mathematical expressions that can be included in the optimization function of the fuzzy rule-based system. Basically, distinguishability can be formally defined as a relation between fuzzy sets defined on the same universe of discourse. In the following, the main approaches that have been proposed to mathematically formalize this interpretability constraint are presented.

**a) - Similarity measure**

Similarity measures approach is the most adopted way to characterize distinguishability constraint (Mencar & Fanelli, 2008). Similarity is usually defined as the degree to which the concepts represented by fuzzy sets belonging to the same universe of discourse are equal (Setnes et al., 1998). More specifically, similarity between fuzzy sets $A$ and $B$ is quantified by a function called *similarity measure* which assigns a similarity value $S$ to the pair of fuzzy sets $(A, B)$. Similarity measure function is given by the following expression:

$$S(A, B) = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \tag{3.5}$$

Where $|.|$ and $\cap$ represent the cardinality of a set and intersection, respectively.

This approach was applied not only to ensure the distinguishability requirement in fuzzy models but also for rule base simplification procedure (Setnes et al., 1998). The main objective of the simplification procedure, which is applied after producing the initial fuzzy rule-based system via a clustering method, is to remove redundancy that may be present in the form of similar fuzzy sets (or non-distinguishable fuzzy sets) that represent compatible concept and replace them by a new fuzzy set representative of the merged fuzzy sets. By replacing the merged fuzzy sets with the new one, the number of fuzzy sets in the rule base decreases and rule base simplification is achieved. Since similar fuzzy sets are merged, the distinguishability in this way is also achieved. A threshold $\lambda$ controls the value of similarity at which the two fuzzy sets $A$ and $B$ are considered similar and thus should be merged. The choice of a suitable threshold value depends on the application and the user-preference of accuracy versus interpretability. Lower threshold value will enforce the distinguishability constraint and results in less fuzzy sets and generally lower accuracy.

In fact, similarity measures have been adopted by many studies to enhance the interpretability in fuzzy models using different learning methods especially evolutionary-based methods such as Genetic Algorithms (Jimenez, Gomez-Skarmeta, Roubos, & Babuska, 2001; Meesad & Yen, 2002; Roubos & Setnes, 2001; Yaochu, 2000), Symbiotic Evolution (Jamei, Mahfouf, & Linkens, 2001), Coevolution (Peña-Reyes & Sipper, 2003). Although similarity measures are well-suited for capturing the characteristics of the distinguishability requirement, this approach was criticized for the calculation needed which is usually computationally intensive (Mencar et al., 2007).

**b) - Possibility measure**

Another approach to quantify distinguishability was based on a possibility measure. It is defined as the degree of applicability of the soft constraint "$x$ is $B$" for $x = A$ (Mencar et al., 2007). The possibility measure can be evaluated using the following expression:

$$\Pi(A, B) = \sup_{x \in U} \min\{\mu_A(x), \mu_B(x)\} \qquad (3.6)$$

A useful interpretation of the expression (3.6) is the extent to which $A$ and $B$ overlap (W. Pedrycz & Gomide, 1998). Possibility measure has some advantages such as the computational efficiency of the calculation procedure comparing with similarity. In addition, it can be used for both ensuring the high distinguishability and high coverage of fuzzy sets (Mencar et al., 2007). Furthermore, and despite similarity measure is different from possibility measure but under mild conditions (such as continuity, normality, and convexity) similarity and possibility are related monotonically, so that small values of possibility implies small values of maximal similarity. In fact, the possibility measure has some good features but it needs more investigation (Mencar et al., 2007).

**c)- Pointwise property approach**

De Oliveira (1999) has proposed a pointwise property to characterize the distinguishability constraint which states that any element $x$ of $X$ will not have simultaneously high membership value in different fuzzy sets defined on $X$. In other word, if an element in $X$ has a high membership value for a fuzzy set, then it must have a low membership value for all the other fuzzy sets. This reasoning was formulated by the following constraint:

$$\forall\, x \in X\colon \sqrt[P]{\sum_{A\in F}\left(\mu_A(x)\right)^P} \leq 1 \; (P \geq 1) \qquad (3.7)$$

Where $P$ is a user-defined parameter that control the strength of the constraint imposed on the fuzzy sets. For the case $P = 1$, the distinguishability constraint is strong whereas the strength of the constraint is reduced for higher values of $P$ and vanished as $P \to \infty$. This constraint is especially effective for controlling the distinguishability in an online training session where it has been used in (De Oliveira, 1999) as the inequality can be checked when just an input is given. In the case where the inequality in (3.7) is violated, a modification of the fuzzy sets will be made through an appropriate learning algorithm (Mencar et al., 2007).

**d)- Pre-defined fuzzy partition**

Actually, there is a special type of partition called “Strong Fuzzy Partition (SFP)” proposed by Ruspini (1969) that satisfies all the semantic constraints to the highest level particularly when the membership functions are also uniform (Gacto et al., 2011). Using this kind of partitions, however, should consider the accuracy of the system. For example, Fazendeiro and de Oliveira (2005) show how it is possible to get more accurate and less interpretable fuzzy rule-based system by breaking the SFP property.

This issue, namely, interpretability-accuracy trade-off, is an important topic and has been especially addressed in the framework of Multi-Objective Evolutionary Algorithms (MOEA). Another interesting approach for maintaining distinguishability property is proposed by Ishibuchi and his co-researchers (see for example (Ishibuchi & Nojima, 2007; Ishibuchi, Yamamoto, et al., 2005) where they suggested pre-defined fuzzy partitions with different level of granularity ranging from 2 to 5 for each input variable. The partitions compose of well-defined and distinguishable fuzzy sets with clear semantic meaning. In the first stage, all the possible rules are generated using these linguistic terms and then a rule and fuzzy set selection procedure are performed using MOEA to choose the most relevant rules and fuzzy sets. In this way, distinguishability constraint is satisfied to the highest level.

### 3.2.3 Complexity-based constraints

Under this category, we review the complexity-based constraints considered in the literature in order to improve the readability of the fuzzy rule-based systems. This complexity can be addressed by minimizing the number of rules and the rule length (the number of fuzzy sets per rule). In the following is a brief description of these two constraints followed by a presentation of the methods applied to satisfy these two constraints.

***a)- Moderate number of fuzzy sets (antecedent conditions)***

To have a readable rule, the number of fuzzy sets in the antecedent part should be as low as possible and should not exceed 7±2 antecedents, which is the number of entities that can be processed by human being for a short time (Peña-Reyes & Sipper, 2003).

***b)- Moderate number of rules***

The number of rules in a fuzzy rule-based system should not be high and the set of rules should be as small as possible under the condition of maintaining the system's accuracy at a satisfied level. A compact (or parsimonious) system is desirable especially in high dimensional problems where the number of generated rules tends to be very high which leads to the lack of global understanding of the system (Mencar & Fanelli, 2008). This constraint is motivated by the same psychological factors as in "*Moderate number of fuzzy sets*" constraint, in addition to the principle of Occam's razor -widely adopted in Artificial Intelligence field- which states that the best model is the simplest one that fits and explains well the system's behaviour (Zhou & Gan, 2008). Simplicity and accuracy are usually two competing modelling objective and should be carefully balanced during the design process (J. Casillas et al., 2003b).

### 3.2.4 Methods applied to achieve the complexity-based constraints

Several methods have been applied to reduce the fuzzy rule base complexity either by reducing the number of rules, fuzzy sets or even both of them. These methods can be grouped into the following categories.

**a) - Reduction of the rule set**

Fuzzy rule set reduction procedure aims at minimizing the number of rules in the rule base while maintaining (or even improving) the accuracy of the fuzzy rule-based system. Actually, there are two approaches applied to reduce the number of rules.

***Rule selection***

The main goal of rule selection procedure is to choose an optimized subset of rules from an initial set of rules. This approach has been widely adopted in the literature to reduce the complexity of the rule base.

Ishibuchi et al. (1995) proposed a genetic algorithm to construct a compact fuzzy rule-based classification system by selecting a small number of linguistic rules with high classification accuracy. This method reduces the complexity of a fuzzy rule-based system thanks to the reduction of the number of fuzzy rules. A comparison was made by Ishibuchi et al. (1997) between a single-objective and two-objective genetic algorithm for finding a set of non-dominated solutions of the rule selection problem. Then, the two-objective problem employed in the previously stated study (the two objectives are the minimization of the number of rules and the maximization of the classification accuracy) was extended to a three-objective problem by including the total number of antecedent conditions as a third objective (Ishibuchi et al., 2001). The aim of the third objective is to build an interpretable fuzzy rule-based system for high-dimensional problems using short linguistic fuzzy rules (incomplete rules). To improve the search ability of the genetic algorithms in achieving the said three objectives and finding the maximum trade-off between the accuracy and interpretability, Ishibuchi, Yamamoto, et al. (2005) proposed a hybrid multi-objective genetic approach which combines Pittsburgh and Michigan fuzzy genetic-based methods that aim to build a compact as well as accurate fuzzy rule-based system by minimizing its complexity and maximizing its accuracy. Ishibuchi and Nojima (2007) further developed the previously cited study by using a well-known multi-objective genetic algorithm called NSGA-II to improve the quality of the generated fuzzy rule-based systems. In addition, they examined the relationship between the accuracy and the interpretability by analyzing the interpretability-accuracy trade-off in the framework of NSGA-II.

Jorge Casillas et al. (2005) proposed a method which integrate the concept of *"linguistic hedges"* in the fuzzy rules. These new linguistic hedges such as *"very small"* or *"more-or-less small"* give the fuzzy rule-based system the ability to achieve a better accuracy while keeping a good interpretability. The authors proposed a genetic tuning process for jointly fitting the fuzzy rule linguistic terms and the meaning of the involved membership functions. The Wang-Mendel method (L. X. Wang & Mendel, 1992) is used first to obtain the initial rule base and then a rule selection together with the tuning of membership functions are performed. The tuning process is applied to fit (match) the changes resulting from the integration of *linguistic hedges* in the fuzzy rules.

The linguistic 2-tuples representation scheme introduced in (F. Herrera & Martinez, 2000) was considered in (Alcala, Alcala-Fdez, & Herrera, 2007) to propose a model of tuning for fuzzy rule-based systems with a rule selection using genetic algorithms. The original fuzzy sets under 2-tuples representation are allowed to slightly displace to the left/right (lateral displacement) which improves the accuracy of the fuzzy rule-based system while maintaining the semantic or the interpretability of fuzzy rules at a reasonable level.

Unlike many existing methods, (Pulkkinen & Koivisto, 2008) employed C4.5 algorithm to create a compact decision tree and then transform the decision tree into a fuzzy classifier. This classifier was utilized to create an initial population for NSGA-II algorithm by replacing some parameters of the classifier to produce a more diverse population. NSGA-II was able to find a set of non-dominated solutions with different accuracy-interpretability trade-off values.

Another interesting approach for rule selection proposed by (Krone & Taeger, 2001) was based on statistical measures which indicate whether a given rule is relevant or not.

In (Yen & Wang, 1999), a selection method based on orthogonal transformation-based methods employs some measure index to detect the most important fuzzy rules that should be retained and less important rules that should be eliminated.

***Rule merging***

Rule merging is an alternative approach for rule set reduction by merging the existing fuzzy rules to create a more compact fuzzy rule set. Usually, two or more rules can be merged if they have the same conclusion and their premises can be merged (J. M. Alonso, 2007). In (Klose, Nürnberger, & Nauck, 1998), a method was proposed to merge neighboring rules where the linguistic terms used by the same variable in each rule are adjacent using three different levels:

1- Rules with neighboring fuzzy sets can be merged using a new fuzzy set that represents the merged fuzzy sets.

2- Rules can be merged on a logical level using disjunctive normal form.

3- Similar fuzzy sets can be merged using a similarity measure.

Another approach was applied on Takagi–Sugeno-type fuzzy rule-based systems (Roubos & Setnes, 2001; Setnes et al., 1998; Setnes & Roubos, 2000) where fuzzy rule set reduction is achieved indirectly as a result of a simplification procedure that aims to reduce the number of fuzzy sets by merging similar fuzzy sets into new one that represents the merged fuzzy sets. This procedure may result in equal rules in the rule base and only one of the equal rules is needed and the others can be deleted.

***Controlling the granularity of the fuzzy partition***

The number of fuzzy sets in a variable (granularity) should be moderate, preferably not more than 7±2. This criterion is defined by a psychological study reported in (Miller, 1956), which found that the number of entities that can be perceived, processed and remembered by a human being for a short time is around 7, plus or minus 2. This finding has been widely adopted by a number of studies related to interpretable fuzzy modelling (De Oliveira, 1999; Espinosa & Vandewalle, 2000; Y Jin et al., 1999; Peña-Reyes & Sipper, 2003; Setnes et al., 1998; Zhou & Gan, 2008). In addition, this study provides a scientific explanation to a common criterion for fuzzy interpretability which states that reducing the complexity of a fuzzy rule-based system leads to enhance its interpretability (Mencar & Fanelli, 2008).

Because the granularity of the fuzzy partitions (i.e. the number of fuzzy sets per variable) influences proportionally the number of rules, some authors (Alcalá, J., Herrera, & Otero, 2007; O. Cordón et al., 2001; Cordon et al., 2001) proposed some methods to decrease or control the complexity of the rule base by finding the optimum number of fuzzy sets for each variable.

**b) - Reduction of fuzzy sets**

Actually, there are two approaches that have been applied for fuzzy rule base simplification (i.e. fuzzy sets reduction): the first is fuzzy sets selection where only the most relevant fuzzy sets are selected and second is fuzzy sets merging where similar fuzzy sets are merged. The aim of fuzzy sets reduction is to produce incomplete fuzzy rules (which contain only the most influential antecedent conditions) because these rules are shorter and easier to understand than complete rules (Gacto et al., 2011).

***Fuzzy set selection***

Fuzzy sets selection procedure was applied generally using multi-objective genetic algorithms where minimizing the number of antecedent conditions is set as one of objectives that should be achieved in the rule base (Ishibuchi et al., 2001; Ishibuchi & Nojima, 2007; Ishibuchi, Yamamoto, et al., 2005; Narukawa, Nojima, & Ishibuchi, 2005). In addition, decision tree method was utilized to generate incomplete rules by performing local feature selection (i.e. fuzzy sets selection) at the rule level (J. M. Alonso & Magdalena, 2011a; José M Alonso et al., 2008; Mikut et al., 2005; Pulkkinen & Koivisto, 2008).

***Fuzzy set merging***

Merging fuzzy sets was applied to simplify the rule base using different methods such as similarity measures (Roubos & Setnes, 2001; Setnes et al., 1998; Setnes & Roubos, 2000), approximate similarity measures (Chen & Linkens, 2004) or using distance measure (Espinosa & Vandewalle, 2000). For the latter measure, the authors stated that two fuzzy sets can me merged when the modal values of their membership functions are “too close” to each other. A sophisticated distance function with internal and external distances to merge the fuzzy sets was proposed in (S. Guillaume & Charnomordic, 2003, 2004).

***Reduction of both the number of rules and fuzzy sets***

Feature selection methods have been used for reducing the complexity of the rule base especially for high dimensional problems because the reduction of the dimension or the input variables will lead to the decrease in the number of generated rules as well as the number of the antecedent conditions for each rule. Feature selection is a pre-processing step that aims at reducing the number of features by selecting the most relevant ones that can produce the best model according to a certain criterion. This criterion is usually minimizing the number of features and maximizing the accuracy of the model.

A feature ranking algorithm was proposed by Tikk, Gedeon, and Wong (2003) adapted to fuzzy classification modelling in order to reduce the complexity of the model. This method applied a clustering method to the data output and then used the cluster-membership degrees as weights in the feature ranking method. In addition, they applied the sequential backward selection (SBS) search method (Devijver & Kittler, 1982) to determine the feature ranking.

Another feature selection proposed by Vanhoucke and Silipo (2003) ranks inputs features according to their mutual information and discards all the irrelevant ones using a threshold criterion. The best criterion, defined after investigating several strategies, is based on discarding all features above a given percentile in the mutual histogram across inputs.

This method has been applied in speech recognition problem and specifically to select the most relevant speech input features to classify the speech segments into their respective phonetics properties.

For some studies (José M Alonso et al., 2008; Mikut et al., 2005; Pulkkinen & Koivisto, 2008), feature selection procedure was applied using the popular decision tree algorithm C4.5 (J. R. Quinlan, 1993) or the decision tree algorithm ID3 (J. Ross Quinlan, 1986) where a feature selection procedure takes place implicitly during the induction of decision tree and rule pruning.

For example, by using a decision tree induction method, Mikut et al. (2005) proposed an interpretable fuzzy rule-based system for classification problem where the rules are pruned in order to obtain simple rule conditions (premises). The information entropy measures are used for selection and pruning which allow the user to control the trade-off between accuracy and interpretability. The main advantage of the proposed method is its ability to produce a compact rule base by using only the locally most relevant features to generate incomplete rules (Serge Guillaume, 2001) but the drawback of this approach is its over-sensitive to noise, outliners or irrelevant attributes (Gacto et al., 2011).

Another approach, which is widely used, employs multi-objective genetic algorithms or specifically designed GA such as SGERD (Mansoori et al., 2008) to minimize the number of fuzzy sets and fuzzy rules as in (Ducange, Lazzerini, & Marcelloni, 2010a; Ishibuchi et al., 2001; Ishibuchi & Nojima, 2007; Ishibuchi, Yamamoto, et al., 2005; Mansoori et al., 2008).

## 3.3 Interpretability evaluation in fuzzy rule-based systems

Interpretability evaluation is an important step that aims to compare between different fuzzy rule-based systems in order to choose the most interpretable one. Since interpretability constraints define the characteristics of interpretable fuzzy rule-based systems, they have been used to assess the interpretability by verifying to what degree these constraints are valid for a given system (Mencar, Castiello, Cannone, & Fanelli, 2011). Some approaches for interpretability evaluation were used to assess the semantic-based constraints (low-level constraints), which are related to the fuzzy sets and fuzzy partition. In many other cases, semantic-based interpretability is limited to the evaluation of the distinguishability constraint using different methods such as similarity measures (Roubos & Setnes, 2001; Setnes et al., 1998; Setnes & Roubos, 2000), Pointwise property approach (De Oliveira, 1999), Possibility measure (Mencar et al., 2007), etc. One disadvantage of semantic constraints-based evaluation is the lack of a general and widely accepted way or measure to evaluate the interpretability semantic-constraints.

Complexity-based approach is the commonly used method for interpretability evaluation; its advantage is the use of widely accepted measures that are usually used to assess the complexity of the systems. This approach can be useful especially when the semantic constraints are fully satisfied by the system. For example, (Ishibuchi, Yamamoto, et al., 2005) used the terms of number of rules, total rule length and average rule length to measure the interpretability. They ignore the semantic evaluation as they assume that the semantic constraints are highly fulfilled because they produce rules with pre-defined linguistic terms with clear semantic meaning. In another study, (Marquez, Marquez, & Peregrin, 2010) employed the three metrics to measure the interpretability: (1) the total number of rules, (2) the number of rules with weight associated, and (3) the average number of firing rules. Even though complexity-related measures are easy for calculation and widely accepted, they may not be always suitable for comparison between different fuzzy rule-based systems because they evaluate the interpretability with more than one criterion (for example number of rules, number of antecedents, etc.) which makes, in some cases, finding the most interpretable fuzzy rule-based system more difficult.

The main drawback of either selecting semantic-based constraints or complexity-based constraints separately is the need to assess the overall interpretability of the fuzzy rule-based system for comparison purpose and the need to combine all semantic and complexity-based constraints into one interpretability index like the case of classification accuracy (Zhou & Gan, 2008).

Table 3.1 list of proposed methods and the constraints used to preserve the interpretability of fuzzy rule-based system

| References | | **Interpretability constraints** | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | **Complexity-based constraints** | | | | | | **Semantic-based constraints** | | | | |
| | MOEA | FS | Rule reduction | | Fuzzy sets reduction | | CG | Fuzzy partition | | Fuzzy rules | Coin | Pred |
| | | | RS | RM | FSet.S | FSet.M | | Dis | Cov | Cons | | |
| Ishibuchi et al. (1995) | | | | | | | | | | | | |
| Ishibuchi et al. (1997) | | | | | | | | | | | | |
| De Oliveira (1999) | | | | | | | | | | | | |
| Ishibuchi et al. (2001) | | | | | | | | | | | | |
| Cordon, Del Jesus, Herrera, Magdalena, and Villar (2003) | | | | | | | | | | | | |
| Nauck (2003) | | | | | | | | | | | | |
| Peña-Reyes and Sipper (2003) | | | | | | | | | | | | |
| Tikk et al. (2003) | | | | | | | | | | | | |
| S. Guillaume and Charnomordic (2004) | | | | | | | | | | | | |
| Ishibuchi and Yamamoto (2004) | | | | | | | | | | | | |
| Jorge Casillas et al. (2005) | | | | | | | | | | | | |
| Narukawa et al. (2005) | | | | | | | | | | | | |
| Mikut et al. (2005) | | | | | | | | | | | | |
| Ishibuchi and Nojima (2007) | | | | | | | | | | | | |

| References | | Interpretability constraints | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Complexity-based constraints | | | | | | Semantic-based constraints | | | | |
| | MOEA | FS | Rule reduction | | Fuzzy sets reduction | | CG | Fuzzy partition | | Fuzzy rules | Coin | Pred |
| F. Liu, Quek, and Ng (2007) | | | ■ | ■ | | ■ | | | | | | |
| Mencar et al. (2007) | | | | | | | | ■ | | | | |
| José M Alonso et al. (2008) | | | | ■ | | ■ | | | | | | ■ |
| Pulkkinen and Koivisto (2008) | ■ | | | ■ | | ■ | | ■ | | | | ■ |
| Pulkkinen and Koivisto (2008) | ■ | | ■ | | ■ | | | | | | | ■ |
| J. M. Alonso, Magdalena, and González-Rodríguez (2009) | ■ | | ■ | | ■ | | | | | | | ■ |
| Pulkkinen and Koivisto (2008) | ■ | | ■ | | ■ | | | | | | | ■ |
| J. M. Alonso et al. (2010) | | ■ | ■ | | ■ | | | | | | | ■ |
| J. M. Alonso and Magdalena (2011a) | | ■ | ■ | | ■ | | | | | | | ■ |
| Gacto et al. (2010) | ■ | | ■ | | | | | ■ | ■ | | | |
| Mencar et al. (2011) | | | | | | | | | | | | |
| (Ishibuchi & Nojima, 2013) | ■ | | ■ | | ■ | | | | | | | ■ |
| (M. Antonelli et al., 2014) | ■ | | | | | | | ■ | ■ | | | |

Note: MOEA: Multi-Objective Evolutionary Algorithms, FS: Feature selection, RS: Rule selection, RM: Rule merging, FSet.S: Fuzzy set selection, FSet.M: Fuzzy set merging, CG: controlling the granularity, Dis: Distinguishability, Cov: Coverage, Cons: consistency, Coin: Cointension, Pred: Predefined fuzzy sets

***Interpretability index***

Nauck (2003) is the first to introduce the index by combining the complexity of a classifier, the number of labels (linguistic terms) and the coverage degree of the fuzzy partition. The Interpretability index is the product of the three following terms:

**Comp:** it represents the complexity of the fuzzy rule-based system and it is calculated as the number of classes divided by the total number of antecedent conditions.

**Conv:** denotes the average normalized coverage and it measures the degree of coverage provided by the fuzzy partition.

**Part:** it denotes the average normalized partition index for all the input variables used in the system. This index is used to penalize partitions with a high granularity.

This index, which is known as Nauck's Index, is further improved by José M Alonso et al. (2008), in which they proposed a fuzzy Index instead of numerical index to measure the interpretability using six main inputs and one output, which is the interpretability. The inputs are: (1) the number of rules, (2) total number of antecedent conditions, (3) number of rules which use one input, (4) number of rules which use two inputs, (5) number of rules which use three or more inputs, and (6) total number of labels defined by input. The inputs are grouped into four linked knowledge bases to form a hierarchical fuzzy rule-based system. The output of the system, which is the interpretability index, composes of five linguistic labels: very low, low, medium, high, and very high. In addition, the authors assumed that fuzzy rule-based systems evaluated include only SFPs so that all the semantic-based constraints are satisfied to the highest level.

In J. M. Alonso et al. (2009), an experimental analysis in the form of a web poll was carried out to evaluate the most used indices for interpretability evaluation. The five indices are: the number of rules (NOR), total rule length (TRL) or the total number of antecedent conditions, average rule length (ARL) or the number of antecedent conditions per rule, Nauck's index (Nauck, 2003) and fuzzy index (José M Alonso et al., 2008). The results showed that there are some significant differences between naïve and expert users and subjectivity in the assessment of the indexes which suggests the need to define a new flexible index that can be easily adapted to the problem and the user preferences. In another study, J. M. Alonso and Magdalena (2010) proposed an index adaptable to the context of each problem by incorporating the user's preferences in the interpretability of the evaluation of a fuzzy rule-based system.

### 3.4 Interpretability vs. accuracy

Ideally, it is better to produce a fuzzy rule-based system which is at once highly interpretable and highly accurate. Unfortunately, this case is not applicable as these two properties are conflicting modeling objectives and thus a highly interpretable system implies a less accurate one and vice-versa. The solution is then to find a good trade-off between them (Cordon, 2011).

The literature shows the existence of two approaches that have been applied to improve the trade-off between the interpretability and accuracy. The first approach generally uses Takagi-Sugeno-Kang (TSK) fuzzy systems to produce an accurate system, and then imposes some constraints to the structure of the system in order to enhance the interpretability by trading-off some accuracy for more interpretability (J. Casillas et al., 2003a). This can be achieved for example by ensuring the distinguishability constraints and reducing the number of fuzzy sets usually through similarity measures (Roubos & Setnes, 2001; Setnes et al., 1998) or reducing the number of the rule base by applying for example orthogonal transformations (Yen & Wang, 1999).

In the other approach, where interpretable Mamdani fuzzy systems are usually employed, some flexibility is added to the system's structure to expand the search space and thus produce more accurate system (J. Casillas et al., 2003a).

Among the strategies applied to enhance the accuracy are: tuning the membership functions by changing the definition values of the parameters or their types (Jorge Casillas et al., 2005; Cordon & Herrera, 1997; Gacto et al., 2010; Gürocak, 1999; Y Jin et al., 1999; Nauck, 2000; Shi, Eberhart, & Chen, 1999), using linguistic modifiers in the rules that allow for more flexibility without losing the interpretability (Cordon et al., 1998; Fernandez, del Jesus, & Herrera, 2010; González & Pérez, 1999; F. Herrera & Martinez, 2000) and learning the granularity of the fuzzy partitions to choose the level which gives more accuracy (Espinosa & Vandewalle, 2000; Gacto et al., 2010).

Another important question is often asked about how can we find the desired level of interpretability for a given application? Or can we evaluate the interpretability systematically and always in the same way?

In (J. M. Alonso et al., 2009), the authors conducted a study about the best index that can be used for evaluating the interpretability in fuzzy rule-based systems. Five different interpretability indexes selected from the literature were given to users with different backgrounds to choose the best among them. The initial results show that it is difficult to design a general index that can be widely accepted, and this finding confirms a previous result which states that the evaluation of the interpretability is a subjective judgment and can be affected by the person's background (J. M. Alonso, 2007). Therefore, it is desirable to enable more interaction between the system and user so he/she can choose, based on his/her needs and objectives, the desirable level of interpretability-accuracy trade-off.

By checking Table 3.1 that lists the proposed works on fuzzy rule-based system interpretability, we can notice easily that multi-objective evolutionary algorithms are the most used techniques to handle the problem of the interpretability, and this is due to the advantages offered by these techniques. In fact, solving the interpretability problem implies finding solutions, in constrained search spaces, that satisfy many criteria, and some MOEA algorithms such as NSGA-II (Deb et al., 2002) have been successfully used to perform this task. They have been used to find non-dominated solutions where different levels of interpretability-accuracy are provided, and the user can choose between them based on the requirements and the objectives of the application.

## 3.5 Ensemble fuzzy rule-based systems

### 3.5.1 Ensemble methods

Ensemble methods are a set of classifiers whose individual decisions are combined in some ways (typically by weighted or unweighted voting) to classify new examples (Dietterich, 2000). Diversity and accuracy are two important criteria that should be considered when constructing ensembles (Chandra & Yao, 2006; X. Wang & Wang, 2006).

Actually, there have been many methods and strategies developed for ensemble methods during the last two decades and they were extensively used to improve the accuracy of single models (C.-L. Liu, 2005) in many domains such as: medicine (Mangiameli, West, & Rampal, 2004; David West, Mangiameli, Rampal, & West, 2005) finance (Leigh, Purvis, & Ragusa, 2002), information retrieval (Elovici, Shapira, & Kantor, 2006; Lior Rokach, Romano, & Maimon, 2008), etc.

In the literature, ensemble methods were also known by other names including Multiple Classifiers (Xu, Krzyzak, & Suen, 1992), ensemble classifiers (W., F., Y., & H., 2003), among others. In addition, there is a variety in the ensemble techniques which results in conducting of several taxonomies as in (L. Kuncheva, 2005) and (L. Rokach, 2009). In the following, we describe the studies that used ensemble methods to improve the classification accuracy of fuzzy rule-based systems.

### 3.5.2 Ensemble fuzzy rule-based systems in the literature

In (Castro, Coelho, Caetano, & Zuben, 2005), an ensemble of fuzzy classifiers is proposed using an immune-based approach to improve the accuracy of the individual fuzzy rule-based systems. The authors employed an immune-based system called Copt-aiNet (Gomes, de Sousa, Bezerra, de Castro, & Von Zuben, 2003), inspired by the immune network principles, to generate accurate and diverse fuzzy classifiers which are two important criteria to design accurate ensemble methods. The results of applying this method on three data sets showed that the ensemble method achieved better accuracy than the single best fuzzy rule-based system.

(Ishibuchi & Yamamoto, 2003) have proposed an ensemble method in which the members are a set of non-dominated fuzzy classifiers whose accuracy and interpretability were optimized using the multi-objective genetic algorithm NSGA-II. The combination method adopted for the ensemble method is the simple majority vote scheme where the class which receives the highest number of votes from the classifier members is declared as the winner. The main motivation behind choosing a multi-evolutionary approach to construct the ensemble members is its ability to generate diverse and accurate solutions. One advantage the authors cited for using ensemble methods whose member are non-dominated fuzzy classifiers is to avoid choosing one single solution from a set of solutions based on their performance on training data set. The authors found that low rates of classifiers in training data set do not always mean low rates in testing data.

In another study, (Ishibuchi & Nojima, 2005) investigated the effects of using NSGA-II with different measures of interpretability on the classification accuracy of the ensemble methods constructed using the best non-dominated solutions. In their study, the authors employed the following three measures separately: (1) the number of fuzzy rules, (2) the total number of antecedent conditions and (3) both the number of fuzzy rules and number of antecedent conditions. In addition, these measures were also used for single-objective genetic algorithms that employed scalar fitness function to produce, for each measure, one fuzzy classifier. The results of this study showed that the accuracy obtained from NSGA-II are better than the individual fuzzy classifiers obtained from single genetic algorithm, and the best ensemble classifiers were obtained from the third measure (which uses both number of rules and number of antecedents). In addition, the effect of ensemble methods on improving the accuracy of individual classifiers depends on data sets. In some data sets, the ensemble methods outperformed the best fuzzy classifier while in others, the best individuals achieved better accuracy.

To increase the diversity of the non-dominated solutions obtained by NSGA-II, (Y. Nojima & Ishibuchi, 2006) proposed an entropy measure that was included in the NSGA-II fitness function. The results showed that the entropy measure helped, in some cases, for achieving a better classification accuracy.

In a different context, (Y. Nojima, Mihara, & Ishibuchi, 2010) designed an ensemble method within the parallel distributed genetic algorithm framework by choosing a single classifier from each sub-population and then combine the selected fuzzy classifiers as an ensemble classifier. In non-ensemble fuzzy classifier, however, only one fuzzy classifier is selected from all the sub-populations. In this study, genetic fuzzy rule selection process strengthened by parallel distribution was, for some cases, 50 times faster than standard non-distributed approach.

A cumulative likelihood measure was proposed by (Cordon, Quirin, & Sanchez, 2008a) to guide the genetic-based selection process of ensemble members from a set of fuzzy classifiers produced by heuristic method proposed by (Ishibuchi, Nakashima, & Nii, 2005). In the first stage, the fuzzy classifiers were built using resampled training sets generated by two ensemble methods, namely, bagging and random sub-space methods and then GAs were applied to select the most suitable fuzzy classifiers to form the ensemble method using the cumulative likelihood measure which, unlike the classification error-based measure, prevents the overfitting problem (Cordon et al., 2008a). The result showed that this selection approach did not only improve the classification accuracy but also reduced the complexity of the ensemble method and thus decreased the computational cost especially for high dimensional problems with a huge number of variables.

In a similar study, Cordon, Quirin, and Sanchez (2008b) used three different methods for feature selection, namely, Random Subspace and two variants of Battiti's MIFS, greedy and GRASP methods. After generating the fuzzy classifiers using the same method as the previous study, GA-based method guided by the cumulative training error measure was employed for ensemble members' selection. The results obtained from applying the proposed approaches on four data sets, which some represent high dimensional problems, are quite promising. One of the main conclusions drawn from this study is that the feature selection methods applied combined with a heuristic fuzzy rule generation to form ensemble classifier method can be a good approach to solve curse of dimensionality problem in large data sets.

In a subsequent study (Krzysztof, Quirin, & Cordon, 2009), the authors extended their previous work by including in the fitness function two choices of diversity measures with the error training measure to select ensemble fuzzy classifiers. In the GA fitness function, the two diversity measures, namely, the difficulty ($\theta$) and the double fault ($\delta$) were separately combined with error-training measure using two methods: lexicographical order-based fitness function (LOFF) and weighted combination fitness function (WCFF). Comparisons were made between cases considering different combinations of diversity measures and combination methods with feature selection methods as shown in Table 3.2. The results showed that the ensemble fuzzy classifiers improved the classification accuracy of single fuzzy classifier, while the genetically selection procedure was able to reduce the complexity of the ensemble fuzzy classifiers; it has generally increased the accuracy of the original ensemble fuzzy classifiers (i.e. without selection).

The authors in (Trawinski, Quirin, & Cordon, 2009) have replicated the experiments conducted in the previous study but with replacing the fitness function of the GA-based ensemble selection process by two criteria, namely, training error and accumulated likelihood. In general, the best results were achieved by either the initial or selected ensemble fuzzy classifiers comparing with the single best fuzzy classifiers especially in the case of high-dimensional problems.

To investigate the effect of using another inductive algorithm called FURIA (Hühn & Hüllermeier, 2009) as a base fuzzy classifier, the authors conducted an exhaustive experimental study in (Trawinski, Cordon, & Quirin, 2011) which involved mainly two kinds of comparisons. The first comparison involved three approaches, namely, FURIA-based fuzzy ensemble method with Bagging, feature selection, and the combination of both of them. The objective of this comparison is to identify the best approach that gives the best performance when considering FURIA as the base classifier of the ensemble method. To benchmark the results of the proposed framework, the best method among the three approaches, which is FURIA-based fuzzy ensemble method with Bagging, is used in a subsequent comparison against the following methods: Bagging C4.5 ensemble method, random forest and Ishibuchi-based ensemble method. The results showed that FURIA-based fuzzy ensemble method with Bagging was competitive if not better than the other compared methods.

In another study (Trawinski., Cordón, & Quirin, 2012), the authors applied NSGA-II, a well-known multi-objective genetic algorithm, as a selection method that can properly reduce the number of classifiers in the ensemble method while maintain the classification accuracy. Five different two-objective fitness functions that combine four measures, namely, the training error, complexity which is the number of selected classifiers, and two diversity measures: difficulty ($\theta$) and double fault ($\delta$) were applied for GA-based selection process. The results of the application of these measures on 20 data sets showed that the fitness function that combine the training error and diversity achieved better accuracy. In addition, the selected ensemble fuzzy classifiers which achieved the best results in the previously stated comparison outperformed other benchmark methods such as Random forest and Bagging C4.5 ensemble methods in terms of accuracy and complexity.

To increase the diversity of members in the ensemble fuzzy classifier, (Trawinski, Cordon, & Quirin, 2013) adopted Random Linear Oracle (RLO), an ensemble method proposed by (L. Kuncheva & J. J. Rodriguez, 2007). In RLO, each of the ensemble members is replaced by two classifiers and an oracle whose task is to select which of the two classifiers will be used when an instance is presented for classification. An oracle is a random function which randomly splits the feature space into parts (L. Kuncheva & J. J. Rodriguez, 2007). To test whether this method improves the accuracy or not, the authors carried out a comparison between Bagging fuzzy ensemble method with and without RLO. The results obtained from applying the said methods on 29 high complexity data sets from UCI and supported by statistical tests showed that an RLO-based Bagging fuzzy method achieved a significantly higher accuracy. In addition, in another comparison, RLO-based Bagging fuzzy classifier was found to be competitive with other classical RLO-based Bagging ensemble classifiers such as C4.5 and Naïve Bayes.

In their subsequent study, (Trawiński, Cordón, Quirin, & Sánchez, 2013b) have replaced RLO with a new Oracle Random approach called Random Spherical Oracle (RSO). In this approach, the feature space is divided into two parts using an oracle based on random hypersphere (Rodríguez & Kuncheva, 2007). In addition, a GA-based selection method is adopted using the NSGA-II algorithm.

In a novel deployment of fuzzy approach in the ensemble methods, (Trawinski, Cordon, Sanchez, & Quirin, 2013) proposed the use of a fuzzy rule-based system as a combination method for ensemble fuzzy classifier. The proposed method can also be used by any other ensemble methods whose base classifier can produce a certainty grade associated with each class in the data set. This combination method has the advantage, comparing with others, of being transparent, that is; it gives some insights in the way the combination is made because of the use of fuzzy rules which have the ability to describe the combination process. In addition, by employing the GA to automatically define the fuzzy-based combination method; this approach is capable to jointly perform classifier selection and fusion. The proposed method was applied on 20 data sets and the results were competitive compared with other methods.

In order to eliminate the problem of selecting the complexity of the fuzzy rule-based system a priori in the previous study (Trawinski, Cordon, Sanchez, et al., 2013), the authors have employed in (Trawinski, Cordon, & Quirin, 2014), NSGA-II algorithm to automatically derive a classifier ensemble fuzzy combination and provide several fuzzy rule-based classifier ensemble designs with different accuracy-complexity trade-offs in a single run.

### 3.5.3 Interpretability in ensemble methods

In machine learning community, and especially with the recent developments in data mining and knowledge discovery fields, there has been a growing interest in proposing algorithms that produce interpretable models.

To make ensemble methods more attractive classification algorithms, some researchers have been trying to make them more interpretable without losing too much accuracy.

In his study, (Domingos, 1998) proposed a meta-learning method, called Combined Multiple Models (CMM), to extract comprehensible decision tree rules from the ensemble method Bagging. The main idea of this method is to give C4.5, the base learning algorithm adopted for this study, a new set of training data that includes in addition to the original training data, a large number of examples generated and classified according to the ensemble method. The results reported showed that CMM was able to retain on average 60% of the accuracy obtained by the ensemble method relative to a single C4.5 trained on the original data only. In addition, this method produced a decision tree whose complexity is a small multiple (2-6) of the original decision tree.

In (Ferri, Hernández-Orallo, & Ramírez-Quintana, 2002), the authors consider the ensemble methods as an oracle, and only one ensemble member is selected based on its similarity to the ensemble methods. In this case, the selected ensemble member should be, semantically, similar to the ensemble methods, and the similarity metric is evaluated using random data set. This method is applicable only if the single model or the base learning algorithm itself is interpretable as in the case of decision tree.

Wall, Cunningham, Walsh, and Byrne (2003) presented a method that can be used as a medical decision support system to provide an explanation of the ensemble output on a case-by-case basis. Unlike Domingos' method which extracts the rules from the ensemble method to capture its global behavior, in this method, a set of rules is generated locally from each ensemble member. When a new case is introduced for explanation, the rules that are activated by this case will be selected and then ranked according to a fitness function and the winner rule which receives the highest fitness value is selected for explanation.

The main concern related to this approach is the fidelity of the extracted rules which means the ability of these rules to faithfully represent the behavior of the ensemble output. As reported by (Domingos, 1998), there is a trade-off between the fidelity and comprehensibility; rules with high fidelity tend to be quite complex.

SLIPPER is a rule learning algorithm which uses the confidence-rated boosting concept to build the ensemble of rules. In their study on the generalization of Adaboost, Schapire and Singer (1999) proposed the idea of assigning a real-valued confidence to each prediction performed by the base classifiers. This idea was extended by (Cohen & Singer, 1999) to construct weighted rule sets. The drawback of this method is that the label assigned to an instance depends on the vote of many rules and not on a single rule like decision tree, which makes it more difficult to understand the decision compared to the simple decision tree where every instance is labelled by only one disjoint rule (Triviño-Rodriguez, Ruiz-Sepúlveda, & Morales-Bueno, 2008).

In (Freund & Mason, 1999), a new presentation for classification rules, called Alternating decision tree (ADTree), was proposed to improve the interpretability of the boosting method. This presentation is a generalization of decision trees, voted decision trees and voted decision stumps. In this method, a single alternating tree is produced rather an ensemble of trees as in AdaBoost. But unlike decision trees, ADTrees are a scoring classification system where the classification that is associated with the path is not the label of the leaf as in decision trees, instead, it is the sum of predictions along the path. This way of classification causes a lack of interpretability in the same manner as SLIPPER does as the final classification involved several rules (Triviño-Rodriguez et al., 2008).

Table 3.2 list of the proposed ensemble methods for developing fuzzy rule-based systems (FRBSs)

| References | Base classifier | ensemble method | Combination method | Selection method | Data set | Cross-validation method | comparison |
|---|---|---|---|---|---|---|---|
| (Castro et al., 2005) | FRBSs generated using genetic algorithms. | Fuzzy ensemble classifiers were designed using an immune-based method called **Copt-aiNet**. The main motivation behind using this method is its ability to generate diverse and accurate FRBSs. | Simple majority vote (without weights) | Selection method consists of two steps: (1) sorting the ensemble candidates according to their performance using validation data set.<br>(2) The best classifier is considered the first component of the ensemble. Then, the second best classifier is added to the ensemble classifiers. If the performance of the two-ensemble classifier improves, then the addition is confirmed otherwise, the added component is removed. The same procedure is applied to the other components. | Artificial, Bupa, Iris. | Data set was randomly partitioned for training, validation, selection and test. This procedure was repeated 10 times. | - The comparison was between the best FRBS and the ensemble fuzzy classifiers.<br>- The results of the best FRBS obtained on some data set were compared with some works such as (Alves, Delgado, Lopes, & Freitas, 2004) for Bupa data set and (Ishibuchi & Yamamoto, 2004) for Iris data set.<br>- No comparison reported with other ensemble methods. |
| (Ishibuchi & Yamamoto, 2003) | Heuristic rules extraction method was used to generate the candidate rules. | NSGA-II was employed to generate non-dominated FRBSs that form the ensemble fuzzy classifiers. The generated FRBSs were optimized using three objectives. | Simple majority vote (without weights) | No selection was performed (all the members' decisions were considered). | Breast cancer, Diabetes, Glass, Cleveland heart disease, Sonar, and Wine. | 10×10-CV ten independent iterations (with different data partitions). | The comparison was carried out with the results reported in (Elomaa & Rousu, 1999) on the same data sets where six variants of C4.5 algorithm were examined. |
| (Ishibuchi & Nojima, | Heuristic rules | NSGA-II was employed to find non-dominated | Simple majority | No selection was performed (all the members' decisions are | Breast cancer, | 10×2-CV, 2-CV was | The comparison was carried out between: (1) Ensemble fuzzy systems with each |

| References | Base classifier | ensemble method | Combination method | Selection method | Data set | Cross-validation method | comparison |
|---|---|---|---|---|---|---|---|
| 2005) | extraction method was used to generate the candidate rules. | FRBSs. Ensemble FRBSs were formed by choosing 3 or five sets of the best FRBSs. | vote (without weights) | considered). | Diabetes, Glass, Cleveland heart disease, Sonar, and Wine. | executed 10 times. | other where the members were found using three different formulations of evolutionary multi-objective fuzzy rule selection.<br>(2) Individual FRBSs obtained from three different single objective formulations of fuzzy rule selection. |
| (Y. Nojima & Ishibuchi, 2006) | Heuristic rules extraction method was used to generate the candidate rules. | NSGA-II was employed to find non-dominated fuzzy systems. | Simple majority vote (without weights) | No selection was performed (all the members' decisions are considered). | Breast cancer, Diabetes, Glass, Cleveland heart disease and Sonar. | 5×2-CV, 2-CV was executed 5 times. | The comparison was carried out between four formulations of evolutionary fuzzy rule selection.<br>(1) Ensemble fuzzy systems where the components were found using two formulations of evolutionary fuzzy systems.<br>(2) Individual FRBSs obtained from two different single objective formulations of fuzzy rule selection. |
| (Y. Nojima et al., 2010) | Heuristic rules extraction method was used to generate the candidate rules. | Ensemble classifiers were generated from sub-training data sets. | Simple majority vote (without weights) | No selection was performed (all the members' decisions are considered). | Phoneme, Satimage and Pendig. | 3×10-CV (10-CV was executed 3 times). | The comparison was between non-parallel and parallel implementations of evolutionary algorithms with different parameter values. |
| (Cordon et al., 2008a) | FRBSs are produced by a heuristic method proposed by | (1) Bagging<br>(2) Random Subspace<br>(3) Bagging+Random Subspace (RS) | Simple majority vote (without weights) | GA with a cumulative likelihood measure. | Pima, Glass, Vehicle, Sonar | 5×2-cv error, | The comparison was between:<br>(1) Single FRBSs,<br>(2) Ensemble FRBSs (Bagging, RS, Bagging +RS),<br>(3) GA-based selected Ensemble FRBSs |

| References | Base classifier | ensemble method | Combination method | Selection method | Data set | Cross-validation method | comparison |
|---|---|---|---|---|---|---|---|
| | (Ishibuchi, Nakashima, et al., 2005). | | | | | | (Bagging, RS, Bagging +RS). |
| (Cordon et al., 2008b) | FRBSs are produced by a heuristic method proposed by (Ishibuchi, Nakashima, et al., 2005). | (1) Bagging<br>(2) Random Subspace (RS)<br>(3) Two variants of Battiti's MIFS (Greedy and GRASP) | Simple majority vote (without weights) | GA with a cumulative error classification measure. | Pima, Glass, Vehicle, Sonar | 5×2-cv error, | The comparison was between:<br>(1) Single FRBSs with feature selection (RS, Greedy and GRASP),<br>(2) Ensemble FRBSs Bagging with feature selection (RS, Greedy and GRASP)<br>(3) GA-based selected Ensemble FRBSs Bagging with feature selection (RS, Greedy and GRASP). |
| (Krzysztof et al., 2009) | FRBSs are produced by a heuristic method proposed by (Ishibuchi, Nakashima, et al., 2005). | (1) Bagging<br>(2) Random Subspace (RS)<br>(3) Two variants of Battiti's MIFS (Greedy and GRASP) | Simple majority vote (without weights) | - GA with two different fitness functions:<br>(1) difficulty ($\theta$) + training error.<br>(2) double fault ($\delta$) + training error.<br>- Combination methods of the bi-criteria: (1) lexicographical order-based fitness function (LOFF) and (2) weighted combination fitness function (WCFF). | Pima, Glass, Vehicle, Sonar | 5×2-cv error, | The comparison was between:<br>- Ensemble fuzzy classifiers selected by the GA using:<br>- LOFF with ($\theta$)<br>- LOFF with ($\delta$)<br>- WCFF with ($\theta$)<br>- WCFF with ($\delta$)<br>- TEFF<br>- Single fuzzy classifiers with feature selection<br>- Ensemble fuzzy classifiers. |
| (Trawinski et al., 2011) | FRBSs are produced by | (1) Bagging<br>(2) Random Subspace | Simple majority | No selection method was applied but sets of ensemble | 21[3] data sets from | 5×2-cv error, | There was an exhaustive experimental study, but we can generally extracted two |

[3] The data sets are: abalone, breast, glass, heart, ionosphere, letter, magic, optdigits, pblocks, pendigits, phoneme, pima, sat, segment, sonar, spambase, texture, vehicle, waveform, wine, yeast.

| References | Base classifier | ensemble method | Combination method | Selection method | Data set | Cross-validation method | comparison |
|---|---|---|---|---|---|---|---|
| | a method called FURIA (Hühn & Hüllermeier, 2009). | (RS),<br>(3) mutual information-based feature selection (MIFS), and<br>(4) random-greedy feature selection based on MIFS and the GRASP approach. | vote (without weights) | classifiers with different sizes were considered for comparison purpose. | machine learning repository were selected. | | main comparisons from this study:<br>(1) The first comparison was conducted between three different approaches:<br>-Ensemble fuzzy classifiers with Bagging only.<br>-Ensemble fuzzy classifiers with feature selection only.<br>-Ensemble fuzzy classifiers with both Bagging and feature selection.<br>(2) The second comparison was between four methods:<br>- The method which achieved the best performance in the first comparison.<br>- bagging C4.5 ensemble method,<br>- random forests, and<br>- Ishibuchi-based fuzzy ensemble classifiers. |
| (Trawinski. et al., 2012) | FRBSs are produced by a method called FURIA (Hühn & Hüllermeier, 2009). | (1) Bagging<br>(2) Random Subspace (RS),<br>(3) mutual information-based feature selection (MIFS), and<br>(4) random-greedy feature selection based on MIFS and the GRASP approach. | Simple majority vote (without weights) | NSGA-II is used for ensemble fuzzy classifiers selection. The following five fitness functions were used:<br>(1) training error + complexity<br>(2) training error + difficulty ($\theta$)<br>(3) training error+ double fault ($\delta$) | 20[4] data sets from machine learning repository were selected. | | The comparison was carried out between five FURIA-based fuzzy ensemble methods selected by NSGA-II algorithm from 50 fuzzy classifiers candidates using five different fitness functions. The five fitness functions were listed in the **"selection method"** column.<br>The method that achieved the best result was compared with the following methods: |

[4] The data sets are: abalone, breast, glass, heart, ionosphere, magic, optdigits, pblocks, pendigits, phoneme, pima, sat, segment, sonar, spambase, texture, vehicle, waveform, wine, yeast.

| References | Base classifier | ensemble method | Combination method | Selection method | Data set | Cross-validation method | comparison |
|---|---|---|---|---|---|---|---|
| | | | | (4) complexity + difficulty ($\theta$)<br>(5) complexity + double fault ($\delta$) | | | - FURIA-based fuzzy ensemble classifiers with fixed ensemble size proposed in (Trawinski et al., 2011).<br>- FURIA-based fuzzy ensemble classifiers compose of the 50 initial classifiers.<br>- Random Forest with fixed size proposed in (Trawinski et al., 2011).<br>- Bagging C4.5 fuzzy ensemble classifiers with fixed size proposed in (Trawinski et al., 2011). |
| (Trawinski, Cordon, & Quirin, 2013) | FRBSs are produced by a method called FURIA (Hühn & Hüllermeier, 2009). | Random Linear Oracle (RLO) ensemble method. | Simple majority vote (without weights) | Selection method adopted in this study is called Random Oracle. The base classifier, in this method, is replaced with a pair of subclassifiers and an oracle whose task is to select one of the subclassifiers whenever an instance is presented for classification. | 29[5] data sets from machine learning repository were selected. | 5×2-cv error, | There were two comparisons: the first was between:<br>- RLO-based bagging ensemble fuzzy classifier and<br>- bagging ensemble fuzzy classifier without random oracle.<br>The second comparison was between:<br>- RLO-based bagging ensemble fuzzy classifier.<br>- RLO-based bagging ensemble C4.5.<br>- RLO-based bagging ensemble Naïve Bayes. |
| (Trawinski, Cordon, Sanchez, et | FRBSs are produced by a method | Bagging fuzzy classifiers using different combination methods. | The use of a fuzzy rule-based | GA-based method. | 20[6] data sets from UCI. | 5×2-cv error, | There were exhaustive experiments and comparison between the proposed method and the state of the art classifier fusion |

[5] The data sets are: abalone, bioassay_688red, coil2000, gas_sensor, isolet, letter, magic, marketing, mfeat_fac, mfeat_fou, mfeat_kar, mfeat_zer, musk2, optdigits, pblocks, pendigits, ring_norm, sat, segment, sensor_read_24, shuttle, spambase, steel_faults, texture, thyroid, two_norm, waveform noise, waveform1, wquality_white.

[6] The data sets are: abalone, breast, glass, heart, ionosphere, magic, optdigits, pblocks, pendigits, phoneme, pima, sat, segment, sonar, spambase, texture, vehicle, waveform, wine,

| References | Base classifier | ensemble method | Combination method | Selection method | Data set | Cross-validation method | comparison |
|---|---|---|---|---|---|---|---|
| al., 2013) | called FURIA (Hühn & Hüllermeier, 2009). | | system as a combination method. | | | | methods, such as Weighted Majority Voting (Louisa & Suen, 1997), average (AVG) (L. Kuncheva, 2004), Decision Templates (DT) (L. I. Kuncheva, Bezdek, & Duin, 2001) combined with selection and fusion methods such as Greedy Forward Selection, Greedy Backward Selection and GA-based method. |
| (Trawiński, Cordón, Quirin, & Sánchez, 2013a) | FRBSs are produced by a method called FURIA (Hühn & Hüllermeier, 2009). | Random Spherical Oracle (RSO) ensemble method. | Simple majority vote (without weights) | NSGA-II was employed to select the members of the ensemble method. | 29[7] data sets from machine learning repository were selected. | 5×2-cv error, | There were two main comparisons: the first is between:<br>- RLO-based Bagging ensemble fuzzy classifiers.<br>- RSO-based Bagging ensemble fuzzy classifiers.<br>The second comparison is between:<br>- RSO-based Bagging ensemble fuzzy classifiers.<br>- C4.5 Bagging ensemble classifiers.<br>- Naïve Bayes ensemble classifiers.<br>- Random forest ensemble classifier. |
| (Trawinski et al., 2014) | FRBSs are produced by a method | Bagging fuzzy classifiers. | The use of a fuzzy rule-based | NSGA-II was employed to generate several fuzzy rule-based classifier ensemble | 20[8] data sets from UCI. | 5×2-cv error, | The performance of FRBCS-CM with NSGA-II was compared with:<br>- Full original ensemble using Majority |

yeast.

[7] The data sets are: abalone, bioassay_688red, coil2000, gas_sensor, isolet, letter, magic, marketing, mfeat_fac, mfeat_fou, mfeat_kar, mfeat_zer, musk2, optdigits, pblocks, pendigits, ring_norm, sat, segment, sensor_read_24, shuttle, spambase, steel_faults, texture, thyroid, two_norm, waveform noise, waveform1, wquality_white.

[8] The data sets are: abalone, breast, glass, heart, ionosphere, magic, optdigits, pblocks, pendigits, phoneme, pima, sat, segment, sonar, spambase, texture, vehicle, waveform, wine,

| References | Base classifier | ensemble method | Combination method | Selection method | Data set | Cross-validation method | comparison |
|---|---|---|---|---|---|---|---|
| | called FURIA (Hühn & Hüllermeier, 2009). | | system as a combination method. | designs with different accuracy-complexity trade-offs. | | | Vote.<br>- Fuzzy rule-based FRBCS-CM[9] from (Trawinski, Cordon, Sanchez, et al., 2013). |

yeast.

[9] FRBCS stands for: fuzzy rule-based classification system, and FRBCS-CM stands for: FRBCS-based combination method

### 3.6 Concluding remarks

After presenting several issues related to the interpretability in fuzzy rule-based systems and ensemble methods, we would like to highlight some important concluding remarks in the following points:

1- An interpretable fuzzy rule-based system should fulfill the two following sets of constraints:

(a) The fuzzy partition should be readable in the sense that it is possible to interpret the fuzzy sets as linguistic labels that are meaningful for the expert. To meet this property, fuzzy partition should fulfill the following semantic constraints: distinguishability, coverage, the use of normal and convex fuzzy sets.

(b) The fuzzy rule-base, which includes a set of rules, should be as compact as possible (few and short rules) so the user can read and understand the rules easily. This compactness feature, which is related to the complexity of the system, can be achieved by reducing the number of rules and antecedent conditions.

2- The interpretability constraints can be generally categorized into semantic and complexity-based constraints. Semantic constraints are considered as *necessary requirements* because they have a direct relation with the validity of the linguistic labels and the extracted knowledge (De Oliveira, 1999; Espinosa & Vandewalle, 2000; S. Guillaume & Charnomordic, 2011) while the complexity constraints are related to the human information-processing capabilities. According to a psychological study (Miller, 1956), the complexity of a fuzzy rule-based system, either in terms of number of rules or number of antecedent conditions per rules, should not exceed 7±2 which is the number of entities that can be processed by human being for a short time.

3- There is a lack of a general and widely accepted way for measuring the interpretability in fuzzy rule-based systems, unlike the accuracy measurement, despite the efforts made by many researchers to propose measures and indexes for interpretability. This problem makes the interpretability evaluation of fuzzy rule-based systems and their comparison more difficult.

4- There are widely accepted measures to evaluate the complexity constraints such as the number of rules, the number of conditions per rule, but there are still no widely accepted measures for evaluating the semantic-based constraints.

5- Some of the semantic-based constraints, such as: normality, convexity, can be easily satisfied by simply selecting the most common membership function types such as: triangular, Gaussian, trapezoidal functions.

6- There is a special type of partition called "Strong Fuzzy Partition (SFP)" (Ruspini, 1969) that satisfies all the semantic-based constraints (distinguishability, coverage, normality, etc.) to the highest level. SFP is widely assumed to have high semantic interpretability, particularly when the membership functions are also uniform (Gacto et al., 2011). Using this type of partition, however, one should always consider the performance of the fuzzy rule-based system because the accuracy can be decreased if the parameters values are too restrictive.

7- Most of the existing proposals for designing interpretable fuzzy rule-based systems consider only some constraints by focusing either on semantic or complexity constraints. A more attractive approach is to construct a fuzzy rule-based system which considers both types of the constraints such in (Ishibuchi & Nojima, 2007; H. Wang, Kwong, Jin, Wei, & Man, 2005b).

8- Ishibuchi and Nojima (2007) proposed an interesting approach that takes into consideration both the semantic and complexity constraints.

9- The accuracy-interpretability trade-off should consider the user-preferences, i.e. it should be a way to introduce the user-preferences in the selection of the desired level of accuracy-interpretability trade-off.

10- Multi-objective genetic algorithms are effective methods for building interpretable fuzzy rule-based systems because they have the ability to combine different and even conflicting objectives into the objective function and offer different levels of interpretability-accuracy trade-off.

11- Different ensemble strategies were integrated into the design of fuzzy rule-based systems to increase their performance, speed up the optimization process and reduce the complexity for high dimensional problems. These strategies range from simple techniques such as Simple Majority Voting Scheme as in (Ishibuchi & Yamamoto, 2003) to more complicated designs that combine multiple strategies in one ensemble system as in (Trawinski. et al., 2012).

12- Interpretability in ensemble methods was also considered in some studies as in (Domingos, 1998; Ferri et al., 2002) where a set of decision rules, which approximately represented the global behavior of the ensemble methods was extracted and used as an explanation tool for the ensemble methods. Unlike the two previous studies, the rules which were locally extracted from the ensemble members in Wall et al. (2003) and served as a medical decision support system provided a local explanation of the ensemble output on a case-by-case basis using the winner rule strategy.

# CHAPTER 4: THE PROPOSED FUZZY-ENSEMBLE METHOD

This chapter introduces the proposed method to build a fuzzy-ensemble classifier.

In a previous work (Ainon, Lahsasna, & Bulgiba, 2012), we designed a fuzzy rule-based system that can be used for acute myocardial infarction (AMI) diagnosis. The proposed classifier consists mainly of two phases. The first one is a feature selection process that aims to identify the factors that would help AMI diagnosis by reducing the number of attributes and selecting the most relevant ones. To achieve this objective, we adopted a genetic-based approach by generating fuzzy systems with different sets of attributes using Fuzzy C-Means Clustering (FCM) method (Bezdek, 1981). Then, a fuzzy rule-based system is selected based on user-defined interpretability-accuracy criteria.

The second phase aims to further optimize the selected fuzzy system in both interpretability and accuracy by replacing its fuzzy sets defined in the first phase with pre-defined fuzzy sets that have clear linguistic interpretations such as low, high and average. In both phases, a Controlled-elitism NSGA-II algorithm was employed for the optimization process.

In another work (Lahsasna, Ainon, Roziati, & Bulgiba), we proposed a fuzzy rule-based system for coronary heart disease diagnosis. To avoid the computational cost of GAs in the feature selection phase, we applied Sequential Floating Forward Selection (SFFS) method instead of GA-based approach applied in (Ainon et al., 2012). In addition, we used an extended format of fuzzy rules that incorporates the grade of certainty and support at the consequent part of each rule. This format aims to help the physician assesses the certainty and the importance of each rule. Furthermore, an ensemble method was applied to enhance the accuracy of the fuzzy rule-based system.

One limitation of these two proposals is the lack of interaction between the two phases, namely, feature selection and fuzzy system optimization as once the attributes are selected in the first phase, the optimization process is performed using only the selected features. This has the advantage of reducing the computational cost especially when the number of features is high but also limits the search space and thus the number of possible solutions which may result in sub-optimal solutions. This problem is solved in the present study. In addition, the ensemble method proposed in (Lahsasna et al.) is further improved in this study by introducing a better ensemble method design.

As can be seen from Figure 4.1, our proposed method composes of three main phases: the first one aims to build an interpretable and relatively accurate fuzzy system while the objective of the second phase is to construct an accurate ensemble method. The last phase combines the interpretability of fuzzy rule-based system with the accuracy of the ensemble method in one fuzzy-ensemble classification system.

In addition, we describe the six benchmark data sets from UCI Machine Learning Repository that are used to evaluate the effectiveness of the proposed method as well as the parameter specifications for the algorithms used in the experiments.

## 4.1 Phase1: Fuzzy rule-based systems

In this phase, we proposed two variant methods of the work proposed in (Ishibuchi & Nojima, 2007) aiming to improve its ability to handle the problem of accuracy-interpretability trade-off in fuzzy systems. This work was selected because of the promising results it showed comparing with other works. In addition, as we stated in Chapter 3, the semantic-based interpretability measure of the fuzzy rule-based system proposed in (Ishibuchi & Nojima, 2007) is fulfilled because of the use of predefined fuzzy sets that have clear linguistic meanings. This will reduce the interpretability problem into fulfilling the complexity-based measure by producing a less complicated and simpler fuzzy rule-based system. Furthermore, this work, which was done by well-known researchers in the area, is the result of a series of enhancements published in different articles (Ishibuchi et al., 2001; Ishibuchi & Yamamoto, 2004; Ishibuchi, Yamamoto, et al., 2005). These works are one of the most cited and recognized in their

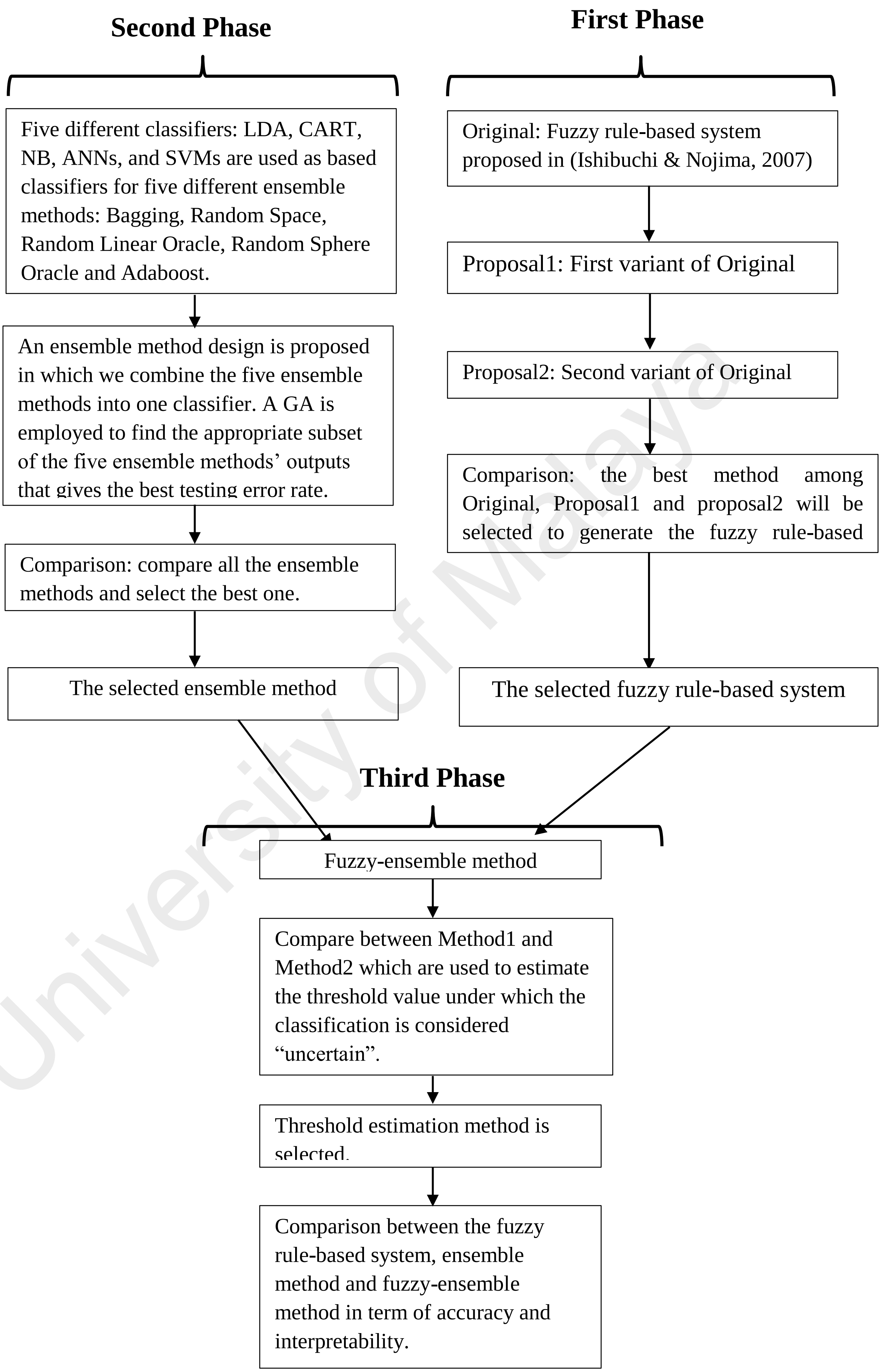


Figure 4.1 An overview of the phases to develop the proposed fuzzy-ensemble method

field[10] and used frequently as benchmark method for result comparisons (Y. Nojima, Takahashi, & Ishibuchi, 2014; Pulkkinen & Koivisto, 2008). Thus, the idea of building on their works and improves their results seems plausible.

Our motivation is that improving the accuracy for fuzzy rule-based systems is an important issue in the research area of fuzzy-evolutionary systems. In addition, the accuracy has an impact even on the interpretability because low accurate rules or more specifically the incorrect classification outputs lead to wrong interpretations.

For convenience, we called the work proposed in (Ishibuchi & Nojima, 2007) Original while the two variants of Original are named Proposal1 and Proposal2. In the following sections, we describe the techniques used for generating and optimizing the fuzzy rules as well the reasoning method adopted in this study. In addition, we give a description of the Original, Proposal1 and Proposal2.

### 4.1.1 Fuzzy rule-based systems for classification problems

In this study, we used fuzzy rules in the following form:

$$R_k: If\ x_1\ is\ A_1^k\ and\ x_2\ is\ A_2^k\ \dots and\ x_n\ is\ A_n^k\ \ Then\ Class\ Y\ is\ C_j\ with\ r^k \tag{4.1}$$

Where $x_1, \dots, x_n$ are linguistic input variables, $A_1^k, \dots, A_n^k$ are linguistic labels representing the values of the linguistic input variables. Every input variable is divided into a finite set of 14 linguistic labels depicted in Figure 4.2. $Y$ is the class $C_j$ in which the pattern $x_p$ belongs, where $j = \{1, \dots, M\}$. So a given pattern $x_p = (x_{p1}, \dots, x_{pn})$ is assigned to one of the $M$ classes. $r^k$ is the certainty grade of the classification in the class $C_j$ .

**Calculate the consequent class and certainty grade**

[10] These three articles in addition to (Ishibuchi & Nojima, 2007) were cited more than 640 times in web of science database and 1000 times in Google Scholar.

A heuristic method (Ishibuchi et al., 1992) is used to determine the consequent class $C_j$ and the certainty grade $r^k$ of the rule $R_k$ in (4.1) as follows:

*Calculate the consequent class*

1- Calculate the compatibility grade $\mu_k(x_p)$ of each training pattern $x_p = (x_{p1}, \ldots, x_{pn})$ with the fuzzy rule $R_k$:

$$\mu_{R_k}(x_p) = \mu_1^k(x_{p1}) \times \ldots \times \mu_n^k(x_{pn}) \qquad (4.2)$$

2- Calculate for each class, the sum of the compatibility grades for the training patterns with the fuzzy rule $R_k$:

$$\beta_{Class\ h}(R_k) = \sum_{xp\ \in Class\ h} \mu_{R_k}(x_p), \qquad h = 1, \ldots, M \qquad (4.3)$$

3- Find class $C_j$ that has the maximum value of $\beta_{Class\ h}(R_k)$:

$$\beta_{Class\ C_j} = \max\{\beta_{class\ 1}(R_k), \ldots, \beta_{class\ M}(R_k)\} \qquad (4.4)$$

We do not generate the fuzzy rule $R_k$ in the case where the consequent class $C_j$ of fuzzy rule $R_k$ cannot be uniquely determined.

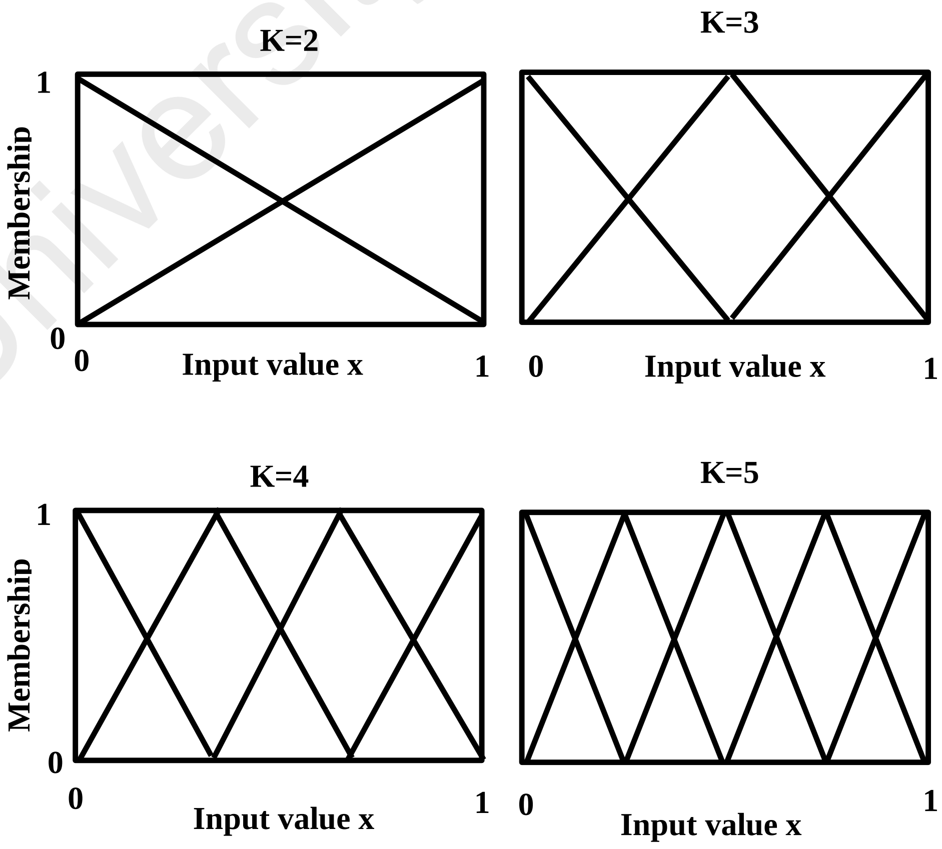


Figure 4.2 Fuzzy partition of the input space. Four fuzzy partitions are used to produce 14 fuzzy sets with triangular shape for each input variable

*Calculate the certainty grade*

4- Calculate the certainty grade $r^k$ as follows:

$$r^k = \beta_{Class\ C_j} - \sum_{\substack{h=1 \\ h \neq C_j}}^{M} \beta_{Class\ h}\ (R_j) \qquad (4.5)$$

If $r^k$ is negative, the rule is not generated.

**Fuzzy reasoning method**

Fuzzy reasoning is the process of inferring a conclusion, or a class label in the case of classification problem, from a set of rules and a pattern. In this study, we adopted the fuzzy reasoning based on a single winner. It is one of the most commonly used method for inferring the class label (Oscar Cordón et al., 1999; Ishibuchi et al., 2001). In this method, the pattern $x_p$ is classified as the consequent class of the winner rule which has the maximum product of the compatibility grade with this pattern and the certainty grade.

Formally, let denote the set of generated fuzzy rules by $R$, where $R = \{R_1, \dots, R_L\}$. When an input pattern $x_p = (x_{p1}, \dots, x_{pn})$ is presented, it is classified as class $C_w$ which is the consequent class of the winner rule $R_w$ and it is determined by the following expression:

$$\mu_{R_w}(x_p).\ r^w = \max\{\mu_{R_k}(x_p).\ r^k \mid R_k \in R\} \qquad (4.6)$$

The classification is rejected in the case where two rules or more have the same maximum value but different classes.

### 4.1.2 Optimizing fuzzy rule-based systems using multi-objective genetic algorithms

After generating all possible rules, the optimization procedure is applied to search for a fuzzy system (i.e. a subset of rules) that has relatively few and short rules but of a high classification ability. These three modelling objectives, namely increasing the accuracy of a fuzzy system, reducing its rules and generating short rules can be formally expressed as follows:

$$Maximize\ f_{acc}(FRBS), Minimize\ f_{rule}(FRBS), Maximize\ f_{dont\ care}(FRBS) \quad (4.7)$$

Where $f_{acc}(FRBS)$: is the accuracy of the fuzzy rule-based system, FRBS, expressed in the number of correctly classified training patterns.

$f_{rule}(FRBS)$: is the number of fuzzy rules in FRBS.

$f_{dont\ care}(FRBS)$: is the number of "don't care" antecedent fuzzy sets conditions in FRBS. The membership of "don't care" is always unity for any input value, i.e., $\forall\ x, \mu_{don'tcare}(x) = 1$. Thus, "don't care" antecedents are not counted. As a result, maximizing the number of "don't care" fuzzy sets produces rules with fewer antecedent conditions or short rules.

Multi-objective genetic algorithms (MOGAs) have been successfully applied in the design of fuzzy rule-based systems for both classification and regression (Fazzolari, Alcala, Nojima, Ishibuchi, & Herrera, 2013).

### 4.1.3 Original method proposed in (Ishibuchi & Nojima, 2007)

Original method is a hybrid method that combines the Michigan approach to find good individual rules (where every one rule is encoded in a single chromosome) with Pittsburgh approach to optimize the rule sets (where a rule set is encoded in a single chromosome).

The outline of this method can be described by the following steps which we call it *main method* of Pittsburgh to distinguish it from Michigan-style method that will be described later in this section[11].

Step1: In the initial population: generate $N$ fuzzy systems.

Step2: Generate the next generations by applying the following operations $N$ times:

(1) Select two fuzzy systems using binary tournament selection.

(2) Generate an offspring from the selected fuzzy systems by crossover and mutation.

(3) Apply a single iteration of a Michigan-style algorithm to the offspring fuzzy system with a probability 0.5.

Step3: Merge the current and offspring populations into one, then choose the best $N$ fuzzy systems for the next generation.

Step4: If prespecified criteria stopping is satisfied, then stop executing the algorithm. Otherwise, return to step2. In the former case, we select all the non-dominated fuzzy systems in the merged population in Step 3 as the final solutions.

**Chromosome design and genetic operators**

Every fuzzy system is encoded in one chromosome which concatenates substrings of length $n$ where each of which represents a single fuzzy rule. The length of a chromosome is not fixed as the number of rules may vary from one fuzzy system to another. The antecedent conditions, in our implementation, are represented with numbers from 1 to 14 and "don't care" antecedent condition is given -1. Figure 4.3 shows an example of a fuzzy system that includes three rules where each one has five antecedent conditions. Rule1 consists only of three antecedents rather than five because there are two "don't care" fuzzy sets which can be omitted. As a result, these two antecedents are not counted in the number of antecedent conditions.

---

[11] For more details about the Original method, readers can refer to (Ishibuchi & Nojima, 2007)

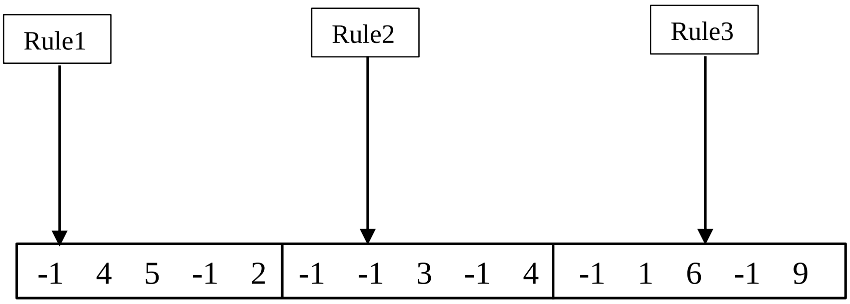


Figure 4.3 Chromosome coding scheme

This fuzzy system composes of three rules and 5 features. The first and fourth features in Rule1 are "don't care" antecedents and thus are not counted

*Initial population*

Every individual or chromosome in the initial population includes 20 rules. In the following is a description of the steps to generate the initial population:

Step1: select randomly, from the training set, one training pattern $x_p = (x_{p1}, \dots, x_{pn})$.

Step2: generate, for the selected training pattern $x_p$, a corresponding rule $R_k$ as follows:

(1) Choose, for each attribute $x_{pi}$, an $A_{ki}$ from one of the 14 candidate fuzzy sets $B_k (k = 1,2,3,4,5,6,7,8,9,10,11,12,13,14)$. For the attribute $x_{pi}$, each candidate fuzzy set $B_k$ has the following selection probability:

$$P(B_k) = \frac{\mu_{B_k}(x_{pi})}{\sum_{j=1}^{14} \mu_{B_j}(x_{pi})}, \quad k = 1, \dots, 14 \qquad (4.8)$$

$B_k$ with the largest probability will be selected as the antecedent condition $A_{ki}$.

(2) Each antecedent $A_{ki}$ is replaced with "don't care" antecedent with probability $P_{don't\ care}$.

(3) Calculate the certainty grade according to the method described in section 4.1.1.

Step3: Repeat Step1 and Step2 20 times to generate one individual or fuzzy system with 20 rules.

Step4: Repeat Step3 $N$ times to create the $N$ individuals (fuzzy systems) of the initial population.

*Crossover operation*

Two parent chromosomes $S_1$ and $S_2$ are selected by the binary tournament selection and numbers of rules $N_1$ and $N_2$ are randomly specified in the interval $[1, |S_1|]$ and $[1, |S_2|]$, respectively. $N_1$ and $N_2$ present the number of selected rules from $S_1$ and $S_2$, respectively.

The new fuzzy system will include $(N_1 + N_2)$ fuzzy rules. When $(N_1 + N_2 > 40)$, only 40 rules will be randomly selected from the $(N_1 + N_2)$ rules. Crossover operation is applied with probability $P_c$. When the crossover is not applied, one of the two parents will be considered as the new offspring.

*Mutation*

Each fuzzy antecedent of the offspring generated from crossover will be replaced with a different antecedent condition using a prespecified probability $P_M$.

**Michigan-style iteration**

Step1: a fuzzy system offspring created from the previous step is handled to the Michigan-style iteration.

Step2: classify the training pattern using the fuzzy system generated from Step2(2) (*Main method*). The fitness value for each rule is the number of correctly classified training patterns by that rule.

Step3: Generate $N_{rep}$ fuzzy rules from the existing fuzzy systems in the population by genetic operations and from misclassified and/or rejected training patterns[12].

Step4: Replace the worst $N_{rep}$ fuzzy rules with the newly generated rules.

Step5: Return the updated rule set of the fuzzy system to the *main method* of Pittsburgh part.

---

[12] There are some details about this step which we purposely skipped. For more details, please refer to (Ishibuchi & Nojima, 2007).

### 4.1.4 Proposal1

In the work described in section 4.1.3, the authors used a multi-objective genetic algorithm called NSGA-II which is an efficient and frequently used genetic algorithm. In this study, we used a variant version called Controlled Elitism NSGA-II. The objective of this replacement is to improve the search ability for finding non-dominated solutions and study the impact of applying a different multi-objective genetic algorithm on both the accuracy and interpretability of the generated fuzzy rule-based systems. In the following is a brief description of the two algorithms, namely, NSGA-II and Controlled Elitism NSGA-II.

**Non-dominated genetic algorithm II (NSGA-II)**

It is an efficient multi-objective genetic algorithms introduced by Deb et al. (2002) to overcome some of the NSGAs (Srinivas & Deb, 1994) drawbacks such as computation complexity, the need for specifying a sharing parameter, and non-elitism approach (Deb et al., 2002).

The advantages of NSGA-II with respect to other multi-objective genetic algorithms is the preservation of diversity and the fast non-dominated sorting of individuals. The concept of non-dominated relation can be defined as follows. Solution $S_A$ dominates $S_B$ if the following two conditions hold (Ishibuchi & Nojima, 2007):

1. $S_A$ is strictly better than $S_B$ in at least one objective, and

2. $S_A$ is no worse than $S_B$ in all objectives.

**Controlled elitist NSGA-II**

It is an enhanced version of NSGA-II proposed by Deb and Goel (2001) for controlling the extent of elitism to a certain portion defined by the user.

NSGA-II selects chromosomes with better fitness value whereas a controlled elitism NSGA-II selects also chromosomes which can help increase the diversity in the population even with lower fitness value. This is done by controlling the elite members of the population as the algorithm progresses. In other words, Controlled Elitism NSGA-II makes a balance between exploitation offered by a selection operator along with an elite-preserving mechanism with exploration offered by a combination operator. This approach allows for a better distribution of solutions and faster convergence comparing with original NSGA-II (Deb & Goel, 2001).

### 4.1.5 Proposal2

In addition to Proposal1 which replaces NSGA-II in the Original method with controlled elitism NSGA-II, we describe in this section another version called Proposal2. In this proposal, we make some changes to the way the chromosomes of the initial population are generated. Specifically, we change Step2 (2) in *initial population* generation related to replacing the selected antecedent conditions with “don’t care” fuzzy sets. Figures 4.4 and 4.5 show the pseudo-code of “don’t care” replacement procedure for Original (and also used in Proposal1) and Proposal2, respectively. In Original method, replacing an antecedent with don’t care takes place only if the probability of a randomly generated number $P_{rnd}$ is more or equal to $P_{dont\ care}$. This procedure is a kind of feature selection where the attributes which are replaced with “don’t care” antecedents are considered as “non-selected features” whereas the others which keep their fuzzy set values as “the selected ones”. In this case, we can say that all the attributes have the same probability to be selected or replaced with “don’t care” antecedents.

In our proposed method, however, we choose a different approach that gives each attribute its own probability of selection based on its importance or relevance. The more the feature is more relevant, the more likely to be selected; or in other words, less likely to be replaced by the “don’t care” antecedent condition. To calculate the probability of features we used the following feature selection procedure.

**for** $k = 1$ *to* 20 **do** ▷ 20 is the number of rules in each fuzzy system
  **for** $i = 1$ *to* $n$ **do** ▷ $n$ is the number of attributes or antecedent conditions in each rule
    $P_{rnd} \leftarrow rand()$; ▷ $rand()$ is randomly generated number between 0 and 1
    **if** $P_{rnd} < P_{dontcare}$ **then** ▷ Check Table 4.2 for $P_{dontcare}$ values
      $Chromo(k, i) \leftarrow -1$; ▷ replace the value in the chromosome with "don't care" value
    **else** ▷ keep the antecedent condition as it is
    **end if**
  **end for**
**end for**

Figure 4.4 Pseudo code for Step2 (2) in initial population generation which replaces some fuzzy antecedent conditions with "don't care" fuzzy sets in Original and Proposal1

**for** $k = 1$ *to* 20 **do** ▷ 20 is the number of rules in each fuzzy system
  **for** $i = 1$ *to* $n$ **do** ▷ $n$ is the number of attributes or antecedent conditions in each rule
    $P_{rnd} \leftarrow rand()$; ▷ $rand()$ is randomly generated number between 0 and 1
    **if** $P_{rnd} > P_{attribute(i)}$ **then** ▷ if random value $P_{rnd}$ is greater than $P_{attribute(i)}$ then replace the value of the $i$ antecedent condition in the chromosome with "don't care" value, i.e. antecedent conditions with lower $P_{attribute(i)}$ are more likely to be replaced with "don't care" antecedents than those with higher $P_{attribute(i)}$
      $Chromo(k, i) \leftarrow -1$; ▷ replace the value in the chromosome with "don't care" value
    **else** ▷ keep the antecedent condition as it is
    **end if**
  **end for**
**end for**

Figure 4.5 Pseudo code for Step2 (2) in initial population generation which replace some fuzzy antecedent conditions with "don't care" fuzzy sets in Proposal2

**Feature selection procedure**

Feature selection is the process of reducing the number of features by selecting a meaningful smaller subset of these features.

Feature selection methods are generally divided into two approaches: wrapper and filters methods (Das, 2001; Kohavi & John, 1997). Wrapper methods are classifier-dependent methods as the selected features are specifically chosen by a particular classifier and thus they are generally more accurate but have some limitations. In addition to their higher computational cost, they are overly specific for the classifier used which likely to render the selected features suboptimal solution for other learning methods (G. Brown, Pocock, Zhao, & Luján, 2012). Filter methods, on the other hand, are classifier-independent, that is, they evaluate the features independently of any particular classifier usually by a scoring criterion, which make them generic and applicable by any classifier. In addition, filter methods are faster and less likely to overfit compared to Wrapper methods (G. Brown et al., 2012).

Filter methods generally rank features according to their individual predictive power using statistical measures (G. Brown et al., 2012). A common approach is to use the Mutual Information between the feature and class label (Gavin Brown, 2009).

Two different feature selection methods may give you different sets of features. In this case, presenting only one set of features selected by a given feature selection method can be misleading (Ludmila I Kuncheva, 2007).

To increase the probability of selecting the appropriate features, we use seven feature selection methods rather than one to produce seven feature set candidates. Then, we calculate the average ranks for each feature over the 7 methods. Finally, we get the probability value of each feature in a way that corresponds to its average ranks. Figure 4.6 shows the pseudocode for calculating the average probability $P_{attribute}(i)$ for each feature. Comparing with the approach used in Original, this method favors features which have good average ranks over the 7 feature ranking methods. This is a kind of guided selection compared to the random selection applied in Original and Proposal1. Factor in the pseudocode listed in Figure 4.6 is an integer which can be used for determining the number of features to be selected. The bigger the value of Factor is, the less the number of antecedent conditions will be selected.

For sonar data set, we make some modifications to our method because the number of features is 60 which is much higher than the other data sets. We calculate the probability of features with the same way as in Figure 4.6 but we apply guided selection on only the top half of the features and a random selection on the remaining features (see Figure 4.7). In addition, we apply also this method, in Step3 of the Michigan style iteration, to the rules generated using Michigan-style approach (in **Michigan-style iteration - Step3**) with probability $P_{Mich}$. The reason behind this hybrid approach for data sets with a very high number of features is that, if we apply guided selection of features only, some of the attributes would never be selected which results in less diverse solutions while applying random selection only may cost a lot of time for the GA to converge to the optimum solutions.

**for** $k = 1$ *to* $7$ **do** ▷ 7 is the number of feature ranking methods
    $ranks \leftarrow feat(k)$ ▷ rank the $n$ features using the $k$ method
    **for** $i = 1$ *to* $n$ **do** ▷ calculate the features probabilities for each $k$ method
        $Probs(k,i) \leftarrow (n - ranks(i) + 1)/(n + (ranks(i) - 1) * Factor);$ ▷ ranks(i) is the rank of the $i$ feature
    **end for** ▷ Factor is an integer (see Table 4.2)
**end for**
$P_{attribute(i)} \leftarrow sum(Probs(:,i))/7;$ ▷ calculate the average probability $P_{attribute(i)}$ over the 7 methods for each feature

Figure 4.6 Pseudo code for calculating the average probability for each feature

**for** $k = 1$ *to* 20 **do** ▷ 20 is the number of rules in each fuzzy system
  **for** $i = 1$ *to* $n$ **do** ▷ $n$ is the number of attributes or antecedent condition in each rule
    **if** $P_{attribute} < P_{mid}$ **then** ▷ $P_{mid}$ is the worst probability of half top features, i.e. all features which have less than $P_{mid}$ do not belong to the half top features
      $P_{rnd} \leftarrow rand()$; ▷ $rand()$ is randomly generated number between 0 and 1
      **if** $P_{rnd} < P_{dontcare}$ **then**
        $Chromo(k, i) \leftarrow -1$; ▷ replace the value in the chromosome with "don't care" value
      **end if**
    **else** ▷ if the feature belongs to the half top features
      **if** $P_{rnd} > P_{attribute(i)}$ **then** ▷ if random value $P_{rnd}$ is greater than $P_{attribute}$
        $Chromo(k, i) \leftarrow -1$; ▷ replace the value in the chromosome with "don't care" value
      **end if**
    **end if**
  **end for**
**end for**

Figure 4.7 Pseudo code for Step2 (2) related to replacing selected fuzzy antecedent conditions with "don't care" fuzzy sets in Proposal2 algorithm for sonar data set

**Feature selection methods**

The following is a brief description of the seven feature selection methods used for ranking the attributes in the data sets. For more details about these methods, please refer to (G. Brown et al., 2012). The authors have also provided a MATLAB toolbox that includes these methods.

***Mutual Information Maximisation (MIM)***

This method, which was applied in (Lewis, 1992), scores each feature independently of others and then ranks the features according to their mutual information. The user will then select a subset of the top ranked features based on some criteria. This method has been frequently used in the literature (G. Brown et al., 2012).

***Conditional Mutual Info Maximisation (CMIM)***

CMIM employs a trade-off between the *relevancy*, or feature power, and *redundancy*, also known by independence, in the search of the most discriminative features. The selection process is performed by iteratively picking features which maximize their mutual information with the predicted class but with the condition that the newly added feature is not similar to the previously selected features (Fleuret, 2004). In other words, the feature which is not carrying additional information, even it is powerful, is not selected.

***Joint Mutual Information (JMI)***

Another approach to reduce the redundancy is by using the *Joint mutual Information (JMI)* measure used by (Yang & Moody, 1999). The idea is to increase the *complimentary* information between features (G. Brown et al., 2012). This method, based on JMI, is simple but effective in reducing redundancy in the features which overcome one of the limitations of Mutual Information approach.

***Double Input Symmetrical Relevance (DISR)***

This method which was proposed in (Meyer & Bontempi, 2006) has used the same concept of complementary information between the features to avoid redundancy as in JMI but used different criterion measure called the *symmetric relevance*.

The method is called *double input symmetrical relevance* (DISR) as it measures the symmetrical relevance on all combination of two features (Meyer & Bontempi, 2006).

***Interaction Capping (ICAP)***

ICAP was proposed by (Jakulin, 2005) and it originates from the idea that a single attribute can be considered irrelevant to the class but when combined with other features, it becomes very relevant. The author proposed to use interaction gain as a measure for detecting attribute interaction. Thus, in the absence of its interacting features, a feature could lose its relevance.

***Conditional redundancy (condred)***

This method was proposed in (G. Brown et al., 2012) for a comparison purpose.

***Conditional Informative Feature Extraction (CIFE)***

CIFE was introduced by (D. Lin & Tang, 2006) based on two key concepts: class-relevance and redundancy. It aims to maximize the joint class-relevant information by reducing the class-relevant redundancies among features (D. Lin & Tang, 2006).

## 4.2 Phase2: Ensemble methods

The second phase aims at constructing an accurate ensemble method that will be used, in the fuzzy-ensemble method, only for prediction. In order to choose the best ensemble method, a series of experiments and comparisons is conducted. The first step in phase2 is to build ensemble methods with different base classifiers. We selected five base classifiers that are known for their classification ability and commonly used to solve classification problems in different domains. In addition, we used five ensemble methods that showed good results compared to others. Furthermore, these ensemble methods can be used with any classifier.

For each of the five base classifiers, we build five different models using five different ensemble methods that are described in the next section. Then, we make comparisons between these models in terms of testing error rates.

The next step in phase2 is to propose a design of an ensemble method in which we combine the five ensemble methods into one classifier. In addition, a multi-objective genetic algorithm is employed to find the appropriate subset of the five ensemble methods' outputs that gives the best testing error rate. In this regard, two different diversity measures, namely, *double fault* and *difficulty* are separately combined with the accuracy measure in the GA's fitness function.

In next sections, we give a brief description of the five ensemble methods followed by the five base classifiers. Then we describe the proposed ensemble method and the GA-based selection method used.

### 4.2.1 Ensemble methods

In this section, we briefly describe five ensemble methods that are used in this study. All these methods have two common features: the use of resampling techniques to create new training sets in order to introduce more diversity and they can be used by any base classifier.

**(a) Bagging**

In this method, the original training data set is manipulated to construct multiple versions of training sets called *bootstrap replicates* of the original data set. Let assume we have a training set *D* with *m* examples, a bootstrap replicate $D_i$ is created by drawing randomly *m* examples from *D* with replacement. Each bootstrap replicate $D_i$ is used to form a classifier $h_i$. This technique is called "**b**ootstrap **agg**regat**ing**" but it is better known by its acronym "**bagging**". The method applied for aggregating the classification decisions of these classifiers is by voting, where each classifier $h_i$ votes for the predicted class, and the class which accumulates the highest number of votes is the predicted class. Figure 4.8 shows the steps applied in training and testing phases for Bagging.

- **Input:**
  - Data set $D = \{(x_1, y_1), \dots, (x_m, y_m)\}$, where $x_i \in X, y_i \in Y$ ; $X = \{\ X_1, \dots, X_n\ \}$.
  - Base classifier $\mathcal{L}$ (such as decision tree algorithm CART);
  - Number of rounds or ensemble members is $T$;

- **Training phase**

for t=1,…,T

(1) Make a boostrap sample by selecting with replacement $m$ instances from $D$.
(2) Train the new classifier $h_t$ using the new $D_t$
end

- **Testing phase**

For each new testing instance $x$,

for t=1,…,T

- Find the class label of $x$ using the trained classifier $h_t$, this label is a "vote" for the respective class.

end

- The class label with the largest number of votes is assigned to $x$.
- Return the ensemble class label for the new instance $x$.

Figure 4.8 Training and Testing phases for Bagging

**(b) AdaBoost**

AdaBoost is one of the commonly used ensemble methods. This algorithm has a wide range of applications (Wu et al., 2008).

As in Bagging, AdaBoost manipulates the training examples to produce multiple training sets. During a series of rounds $t = 1, \dots, T$, it maintains a set or distribution of weights $D_t$ over the training set $D$ = $\{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$, where $x_i \in X$ and $y_i \in Y, i \in \{1, \dots, m\}$. Initially, all the training examples are given the same weights then gradually, on each round $t$; the weights of examples which are incorrectly classified are increased so that the classifier is forced to focus on the examples that are difficult to predict. As it is depicted in Figure 4.9 which describes the steps of the algorithm, in each iteration $t$, a new classifier $h_t$ is constructed using the updated distribution of weights $D_t$. Then, the classifier $h_t$ is assigned, based on the error $\epsilon_t$, a weight $\alpha_t$ which measures its importance. The final classification $H$ is derived by weighted majority voting of the $T$ weak learners.

**(c) Random Subspace**

In random subspace, the training sets, which are used to train each classifier in the ensemble, are made by random selection of a subset of the original features (Lai, Reinders, & Wessels, 2006). The final output of the ensemble method is calculated by aggregating the outputs of the ensemble members using the majority vote method. Random space, like Bagging, can be used with any base classifier. The main idea behind this method is that classifiers which are trained on data sets with different feature subsets may become more diverse which leads to improvement in the accuracy (Ludmila I. Kuncheva & Rodríguez, 2010). Figure 4.10 displays the steps of the training and testing phases of the algorithm.

- **Input:**
  - Data set $D=\{(x_1, y_1), \dots, (x_m, y_m)\}$, where $x_i \in X, y_i \in Y = \{-1, +1\}$ ;
  - Base classifier $\mathcal{L}$ (such as decision tree algorithm CART);
  - Number of rounds or ensemble members is $T$;

- **Training phase:**
  - Set the weight $\mathrm{w}^1 = [\mathrm{w}_1^1, \dots, \mathrm{w}_\mathrm{m}^1]$, $w_j^1 \in [0,1]$, $\sum_{j=1}^{m} w_j^1 = 1$.
    (Usually $w_j^1 = 1/m$ , j=1,…, m)

for t =1, … ,T

- Take a sample $D_t$ from $D$ using distribution $\mathrm{w}^\mathrm{t}$.
- Train a new classifier $h_t$ using the training data $D_t$.
- calculate the weighted ensemble error at step $t$ using:

$\epsilon_t = \sum_{j=1}^{m} w_j^t\, l_t^j,$   % calculate the error $\epsilon_t$ of $h_t$

($l_t^j$=1 if $h_t$ misclassified $x_i$ and $l_t^j$=0 otherwise)

- if $\epsilon_t = 0$, reinitialize the weights $w_j^t$ to $1/m$ and continue.
- else if $\epsilon_t \geq 0.5$, ignore $h_t$, and reinitialize the weights $w_j^t$ to $1/m$ and continue.
- else calculate

$\beta_t = \frac{\epsilon_t}{1-\epsilon_t}, where\ \epsilon_t \in (0,1)$ , then update the individual weights

$$w_j^{t+1} = \frac{w_j^t \beta_t^{(1-l_t^j)}}{\sum_{i=1}^{m} w_i^t \beta_t^{(1-l_t^i)}}, j = 1, \dots, m$$

  - Return the trained classifier $h_t$ and the corresponding weights $\beta_t$.

  end

- **Testing phase:**

For each new testing instance $x$,

for t =1, … ,T

- Classify the instance $x$ using the trained classifier $h_t$

end

  - Calculate the support for the class $C_k$ by

$$\mu_k(x) = \sum_{h_t(x)=C_k} \ln\left(\frac{1}{\beta_t}\right)$$

  - The class with the maximum support is chosen as the label for $x$.
  - Return the ensemble class label for the new instance $x$.

Figure 4.9 Training and testing phases for Adaboost

- **Input:**

- Data set $D$ ={$(x_1, y_1), \dots, (x_m, y_m)$}, where $x_i \in X, y_i \in Y$ ;
  $X = \{ X_1, \dots, X_n \}$.
- Base classifier $\mathcal{L}$ (such as decision tree algorithm CART);
- $d$ (where $d < n$) is the number of features to be sampled for each classifier.
- Number of rounds or ensemble members is $T$;

- **Training phase**

for t=1,…,T

(1) Select randomly without replacement $d$ features from $n$,
(2) Train the new classifier $h_t$ using the new $D_t$ ($D_t$ has $d$ features)
end

- **Testing phase**

For each new testing instance $x$,

for t=1,…,T

(1) For each classifier $h_t$ , select only the features of $x$ that this classifier was trained on.

(2) Find the class label of $x$ using $h_t$, this label is a "vote" for the respective class.

end

- The class label with the largest number of votes is assigned to $x$.
- Return the ensemble class label for the new instance $x$.

Figure 4.10 Training and testing phases for Random Subspace

### (d) Random Linear Oracle and (e) Random Sphere Oracle

For random oracle, the training data for each ensemble member is randomly split into two parts of the space and then one classifier is separately trained on one part (L. I. Kuncheva & J. J. Rodriguez, 2007). For *random linear oracle (RLO)*, the separation is done through a random hyperplane while a random hypersphere separation is applied for *random sphere oracle (RSO)*. Unlike other ensemble methods, each ensemble member is replaced by a mini-ensemble consisting of two classifiers. According to

(Ludmila I. Kuncheva & Rodríguez, 2010), different random splits will create extra diversity in the ensemble while allowing for high accuracy of the ensemble members. These two methods, namely, RLO and RSO, can be used with any base classifier. In addition, they were found to be accurate methods (L. I. Kuncheva & J. J. Rodriguez, 2007; Rodríguez & Kuncheva, 2007). Figure 4.11 shows the training and testing phases for RLO. The same steps are applied for RSO but instead of using hyperplane in Training phase- step (2) and Testing phase- step(1) to split the training data and testing data, respectively, hypersphere in these cases is applied.

- **Input:**
  - Data set $D$ ={$(x_1, y_1), \dots, (x_m, y_m)$}, where $x_i \in X, y_i \in Y$ ;
    $X = \{ X_1, \dots, X_n \}$.
  - Base classifier $\mathcal{L}$ (such as decision tree algorithm CART);
  - Number of rounds or ensemble members is $T$;

- **Training phase**

for t=1,…,T

(1) Generate a random hyperplane (the *t*-th oracle) through the feature space. Store coefficient of the plane equation.
(2) Using the hyperplane, split the training data into two sub-training data $D_{t(1)}$ and $D_{t(2)}$.
(3) Train the two classifiers $h_{t(1)}$ and $h_{t(2)}$ using $D_{t(1)}$ and $D_{t(2)}$, respectively.
(4) Save the trained T×2 classifiers.

end

- **Testing Phase**

For each new testing instance $x$,

for t=1,…,T

(1) Apply the *t*-th saved oracle (random hyperplane) to $x$.
(2) Depends on which hyperplane $x$ belongs, use either $h_{t(1)}$ or $h_{t(2)}$ to get the class label for *t*-th ensemble member.

end

- The class label with the largest number of votes is assigned to $x$.
- Return the ensemble class label for the new instance $x$.

Figure 4.11 Training and testing phases for Random Linear Oracle (RLO)

### 4.2.2 Base classifiers

**(a) Linear Discriminant Analysis (LDA)**

Linear Discriminant Analysis (LDA) was first proposed by Fisher (Fisher, 1936) as a classification method and it has been reported as the most common classification method used in statistics (Sung, Chang, & Lee, 1999). It is available in most of computational software packages.

**(b) Classification and Regression Tree (CART)**

CART is a commonly used decision tree algorithm for both classification and regression problems. It was proposed by (Leo Breiman, Friedman, Stone, & Olshen, 1984). In CART, the models are obtained by recursively partitioning the data space and fitting a simple prediction model within each partition (Loh, 2011). The algorithm of CART is implemented in many software packages including Weka, and MATLAB.

**(c) Naïve Bayes**

Naive Bayes is a simple, probabilistic learning method for constructing classifiers. It is optimal when the features conditionally independent (Ludmila I. Kuncheva & Rodríguez, 2010) but it has been deemed surprisingly accurate even when the independence assumption is clearly false (Hand & Yu, 2001). Naive Bayes is simple and widely used classification method.

**(d) Artificial Neural Networks (ANNs)**

Artificial Neural Networks are computational models inspired by the functioning of the human brain. They are usually composed of basic elements called neurons operating in parallel and arranged in layers. ANNs are widely recognized as powerful learning methods that have been used extensively to solve many problems in different domains. Multilayer Perceptron (MLP) (Haykin, 1998), the most popular supervised neural networks method, is selected as a base classifier for building ensemble methods. A comprehensive package which includes many architectures and learning algorithms of ANNs is included in MATLABs Neural Network toolbox (Demuth & Beale, 2010).

**(e) Support Vector Machines (SVMs)**

SVMs are commonly used classifier methods known for their high classification abilities (Burges, 1998). The most used version of SVMs is the one which used a linear kernel as the discriminant function (Ludmila I. Kuncheva & Rodríguez, 2010). SVMs are originally designed for binary classification but they are also applicable for multi-classification problems. They are implemented in many software packages including IBM SPSS Modeler, Weka and MATLAB.

### 4.2.3 Proposed design for the ensemble method

The most adopted approach to build ensemble methods is to employ only one base classifier. In addition to this approach, we propose a design in which we combine the outputs of all the five base classifiers and then select a subset of the outputs that gives the best performance according a given criterion. Figure 4.12 gives an overview of the ensemble classifier design.

Diversity among the ensemble members is considered as an important issue when their outputs are combined (LudmilaI Kuncheva & Whitaker, 2003). For this reason, we use a multi-objective genetic algorithm called controlled elitism NSGA-II to select a subset of the outputs that maximizes two objectives: accuracy and diversity.

For accuracy measure, we use the number of correctly classified training patterns while two different diversity methods are employed, namely, *double default* and *difficulty*. Since several non-dominated solutions are produced for each training data set, we always select the solution with the best training error rate.

**Initial population**

Every chromosome in the initial population is a binary that randomly created of zeros and ones. Then binary tournament selection was applied to select the parents. We used two-point crossover to create new offspring with different combinations of ones and zeros.

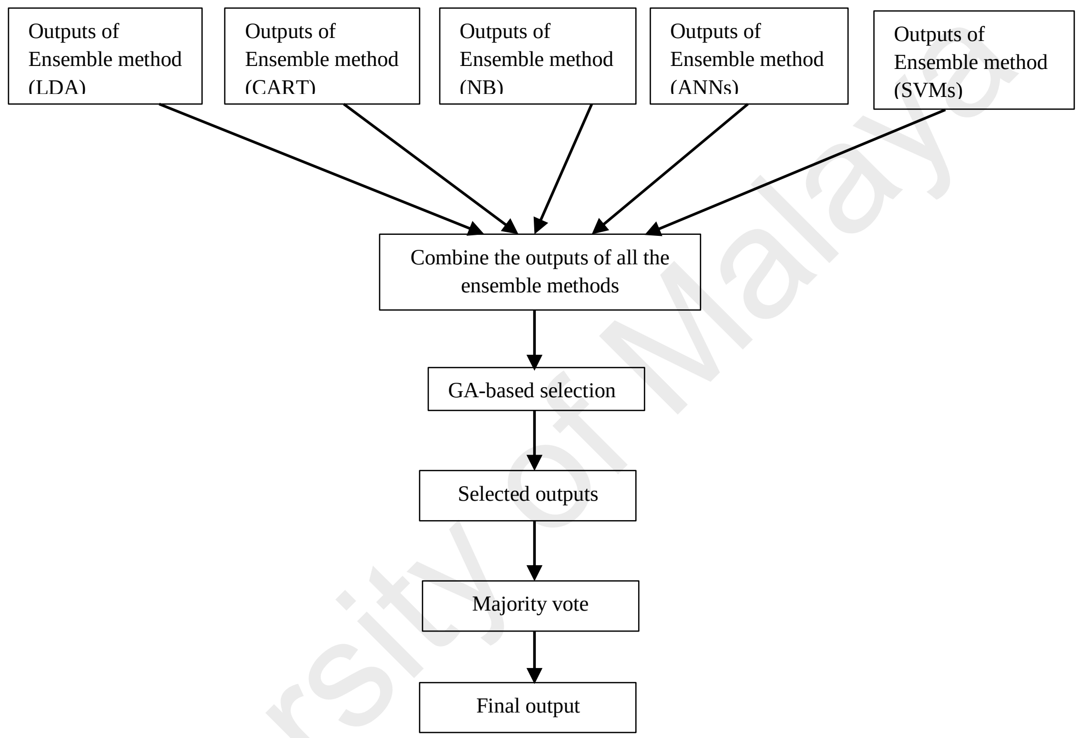


Figure 4.12 An overview of the ensemble classifier design using GA-based selection method

**Fitness function**

The quality of a selected subset of ensemble outputs is defined by two criteria, training accuracy rates and diversity. We can formulate the fitness function for the ensemble output selection with two objectives as follows:

$$\text{Maximize } f_1 \text{ and Maximize } f_2 \qquad (4.9)$$

Where $f_1$ is the accuracy rate of the training data, and

$f_2$ is the diversity measure, which can be either double default ($DF_{av}$) or difficulty ($Diff$).

**Double default**

It was proposed by (Giacinto & Roli, 2001) to select classifiers that are least related. In this measure, a pairwise diversity matrix for $L$ classifiers is formed. Double default (DF) is defined as:

$$DF_{i,k} = \frac{N^{00}}{N^{11} + N^{10} + N^{01} + N^{00}} \qquad (4.10)$$

Where $N^{00}$: the number of patterns which misclassified by both classifiers $i$ and $k$.

$N^{11}$: the number of patterns which correctly classified by both classifiers $i$ and $k$.

$N^{10}$: the number of patterns which correctly classified by classifier $i$ and misclassified by classifier $k$.

$N^{01}$: the number of patterns which misclassified by classifier $i$ and correctly classified by classifier $k$.

Then the average value over the diversity measure is calculated as:

$$DF_{av} = \frac{2}{L(L-1)} \sum_{i=1}^{L-1} \sum_{k=i+1}^{L} DF_{i,k} \qquad (4.11)$$

**Difficulty**

It was proposed by (Hansen & Salamon, 1990) to improve the accuracy of neural networks ensembles. Let $X$ be a discrete random variable taking values in $\{\frac{0}{L}, \frac{1}{L}, \dots, 1\}$ denoting the proportion of classifiers that correctly classify a pattern $x$ drawn randomly from the data (LudmilaI Kuncheva & Whitaker, 2003). To calculate $X$, the outputs of all $L$ classifiers are calculated and difficulty is the variance of $X$.

$$Diff = var(X) \qquad (4.12)$$

**Coding scheme and genetic operator**

Our chromosome is designed as a fixed-length binary vector where the length or number of genes represents the number of outputs of all the ensemble methods. Figure 4.13 shows the coding scheme where the binary values of 1 and 0 indicate that the corresponding output is selected and not selected, respectively. The length of the chromosome is 250 (5 base classifiers×50 classifier members for each base classifier). We applied tournament selection method. The parameter specifications of this method can be found in Table 4.2.

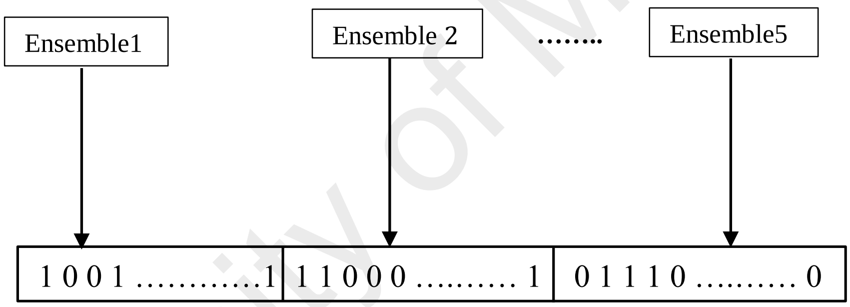


Figure 4.13 Chromosome coding scheme

This chromosome represents the outputs of all the five ensemble methods. Each ensemble method such as Bagging has 50 members and thus 50 outputs, so the total number of all outputs is 250. The value 1 in chromosome means this output is selected and 0 means is not selected.

## 4.3 Phase3: Fuzzy-ensemble method

After selecting the fuzzy rule-based system from phase1 and the ensemble method from phase2, phase3 aims to construct fuzzy-ensemble classifier that combines the strengths of both techniques, namely, the interpretability of fuzzy rule-based system and the accuracy of the ensemble method. But before we explain the proposed method, we introduce in the following subsections some related concepts.

### 4.3.1 Classification rejection

Classification rejection has been studied in pattern classification problems to minimize the cost of misclassification especially when this cost is much bigger than the rejection (Ishibuchi & Nakshima, 1998). The patterns are rejected when they are most likely to be incorrectly classified, and these patterns need to be handled either manually or using more sophisticated methods (Giorgio Fumera et al., 2000).

In non-fuzzy setting and according to Chow's rule (Chow, 1970), a pattern is rejected if the maximum of its a posteriori probabilities is lower than a predefined threshold.

For fuzzy rule-based systems and according to the adopted single winner rule reasoning, the simplest way to use the rejection method of a pattern is when the certainty grade of the winner rule on this pattern is lower than a predefined threshold (Ishibuchi & Nakshima, 1998). This can be formally written by slightly modifying (4.6) equation as:

$$\mu_{R_w}(x_p).\ r^w = \max\{\mu_{R_k}(x_p).\ r^k \mid R_k \in R\} < \theta \quad (4.13)$$

Where $\theta$ is the threshold value under which the classification is rejected.

Another case for classification rejection, as we previously stated in fuzzy reasoning method description (in section **4.1.1**), is when there are two or more rules that have the same maximum value in (4.6) but with different classes.

### 4.3.2 Classification coverage

When a new pattern is presented for classification, the rule which has the maximum product of the compatibility grade with that pattern and the certainty grade (i.e. the winner rule) is used for classification. There are, however, some cases where a pattern is *not covered* by any rule, i.e., the compatibility grade of all the rules is equal to zero. In such case, the pattern is not classified and therefore we need to either classify it manually or use other methods.

### 4.3.3 Global and local interpretability

Fuzzy rule-based systems consist of a set of linguistic rules that can be understood by the human being. For a given classification problem, we can study the rules in order to understand the relationship between the inputs and the outputs. We can as well, in case of short fuzzy rules that include only a few antecedent conditions, determine the important features and their values in relation with the outputs. This kind of understanding or interpretation which gives an overview of the system is considered as a global interpretation because it describes the global behaviour of the system (Nauck, 2000). This interpretability is the standard interpretability known in the literature and it is usually evaluated by the complexity measure of the fuzzy system that includes mainly the number of rules and the number of antecedents per rule.

In addition, we have also the local interpretation that can be used to provide an interpretation to a specific pattern classification (Wall et al., 2003). For example, why this patient was diagnosed as a positive for a given disease?. Or in the case of the credit scoring models which assess the credit worthiness of loan applicants, why this applicant was classified as a good applicant?. The interpretation in these cases are related to the rule which performs the classification or exactly to the winner rule in our case. Both global and local interpretations are important, and can be both needed in one system like in medical decision support systems where the user is interested not only in justifying the diagnosis for a particular patient but also understanding the relationship between the symptoms and the diagnosis outcome, or the most important symptoms related to the disease.

### 4.3.4 The proposed fuzzy-ensemble method

As Figure 4.14 displays, the fuzzy rule-based system is used for both classification and interpretation and only needs the ensemble method support when the certainty grade of its classification is low or in case of rejected and uncovered classifications. But one important question can be asked: do we lose the interpretability of the classification when the ensemble method is used for classification instead of the fuzzy rule-based system?.

Actually, we can get the interpretation of a pattern classified using the ensemble method by identifying the winner rule among the set of rules that has the same class label as the class predicted by the ensemble method. We can justify the logic behind this method as the following: since the classification certainty of the fuzzy rule-based system is below the predefined threshold value then we reject the classification. In this case, we need a reliable classifier to help identify the correct classification which is the ensemble method in our case. So, rather than identifying the winner rule among all the classes, as we do when the classification is performed by the fuzzy rule-based system, we limit the competition only among the rules whose class label is the same as the class predicted by the ensemble method. Thus, the winner rule which has the same class label as the ensemble method is used for local interpretation of the classified pattern. The only case in which we lose the local interpretation of a pattern classification is when the class label produced by the ensemble method is not covered by any rule.

We need to mention that the uncovered patterns do not occur only when we apply the ensemble method but even when we use the fuzzy rule-based system for classification.

**Rejection methods and threshold calculation**

Rejection methods have been used in the literature to define the way in which the classification of a given classifier is rejected (G. Fumera, 2002). The same thing for our case, rejection methods determine when the classification of a fuzzy rule-based system is rejected and thus when the ensemble method is used for classification. We used two commonly rejection methods (Ishibuchi & Nakshima, 1998; Ishibuchi & Nii, 2000) to call the ensemble method. In (Ishibuchi & Nii, 2000), the authors assumed that the threshold is pre-specified. In this study, however, we introduced two methods to calculate the threshold value $\theta$ under which the classification is rejected. These methods (used for threshold value $\theta$ calculation) along with their respective rejection methods will be compared in terms of accuracy and local interpretability rates to choose the most suitable one. In what follows is a description of the two methods.

**Method1:** The threshold $\theta_1$ in this method is calculated as the average product of the compatibility grades and the certainty grades of winner rules that incorrectly classified the training patterns. Threshold $\theta_1$ can be calculated by:

$$\theta_1 = \frac{\sum_{j=1}^{m} \mu_{R_w^j}(x_j).\, r_j^w .\, l_j}{\sum_{j=1}^{m} l_j} \qquad (4.14)$$

Where $r_j^w$ is the certainty grade of the winner rule $R_w^j$ for the training pattern j while $\mu_{R_w^j}(x_j)$ is the compatibility grade of the antecedent part of the winner rule with the pattern $j$. $l_j = 1$ if the fuzzy system misclassified $x_j$ and $l_j = 0$ otherwise. $m$ is the number of training patterns.

The rejection method that corresponds to Method1 works as follows: when a new pattern $x_j$ is presented for classification, we calculate the product $\mu_{R_w^j}(x_j).\, r_j^w$ and then we compare it with $\theta_1$. If $\mu_{R_w^j}(x_j).\, r_j^w > \theta_1$ then we use the fuzzy rule-based system otherwise we reject the classification and call for the ensemble method to perform the classification instead.

**Method2:** in this method, the threshold $\theta_2$ is calculated as the following:

$$\theta_2 = \frac{\sum_{j=1}^{m} (\mu_{R_w^j}(x_j).\, r_j^w - \mu_{R_w^j{}'}(x_j).\, r_j^{w'}).\, l_j}{\sum_{j=1}^{m} l_j} \qquad (4.15)$$

Where $r_j^{w\prime}$ is the certainty grade of the second best rule $R_w^j{}'$ whose class label is different from that of the winner rule $R_w^j$ for the training pattern $j$ while $\mu_{R_w^j{}'}(x_j)$ is the compatibility grade of the antecedent part of the second best rules with the training pattern $j$.

For testing classification, we apply the following rejection method: first we calculate $\mu_{R_w^j}(x_j).r_j^w - \mu_{R_{w'}^j}(x_j).r_j^{w'}$ and then compare it with $\theta_2$. If $\mu_{R_w^j}(x_j).r_j^w - \mu_{R_{w'}^j}(x_j).r_j^{w'} > \theta_2$ then we use fuzzy rule-based system otherwise we reject the classification and call for the ensemble method to classify the pattern $j$.

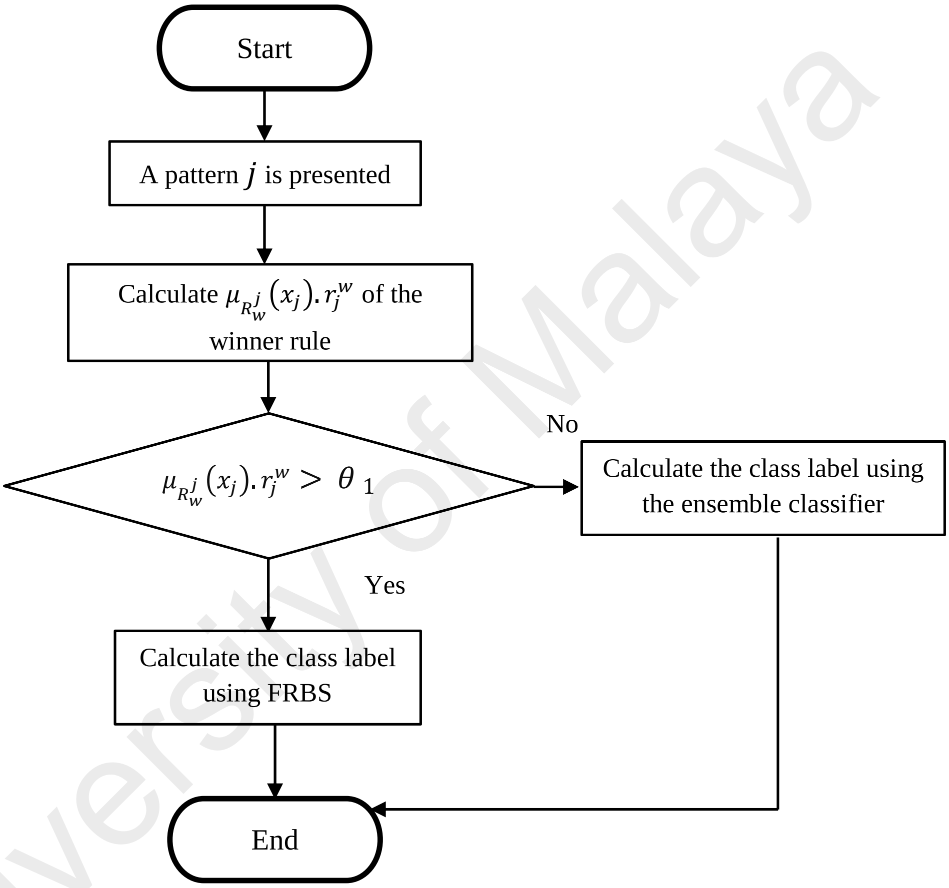


Figure 4.14 Flowchart of fuzzy-ensemble classification with rejection threshold $\theta_1$ associated with Method1

**Comparison between Method1 and Method2**

A comparison between Method1 and Method2 will be made using two criteria: the accuracy and interpretability. The accuracy is evaluated in terms of testing error rates while the interpretability is assessed using the local interpretability rates. We mean by patterns with local interpretability those which are both correctly classified and covered.

In the fuzzy rule-based system, all the *correctly* classified patterns are *covered* by their winner rules and thus have local interpretation. But in case of the fuzzy-ensemble method, some patterns are classified by the ensemble method and they may not have coverage by any rule even if they are correctly classified by the ensemble method. In this case, we cannot get the interpretation of these patterns. In addition, there is no importance to get the interpretation of misclassified testing patterns. So, in order get a local interpretation of a specific pattern classification output, the classification of that pattern has to be both correct and covered. The local interpretability rate can be calculated using the equation:

$$Local\ interpretability\ rate\ (\%) = 100 \times \frac{N_{int}}{m} \tag{4.16}$$

Where $N_{int}$ is the number of testing patterns that are both correctly classified and covered and $m$ is the total number of testing patterns. Equation (4.16) suggests that the local interpretability rate of the fuzzy-ensemble method is less or equal to the testing accuracy rate while they are equal in the case of the fuzzy rule-based system.

## 4.4 Data sets

Six data sets, which are publicly available at the UCI Repository of Machine Learning Databases, are selected for our method evaluation. They have been extensively used in many studies to evaluate the performance of new proposal algorithms as in (Ishibuchi & Nojima, 2007; Pulkkinen & Koivisto, 2008). In addition to the feature of being taken from different classification problems, these data sets have also some desirable features like including data sets that have different number of features, from 8 to 60, and different sizes, from 178 to 768 patterns. Furthermore, we take exactly the same data sets as in (Ishibuchi & Nojima, 2007) for the purpose of comparison. The main characteristics of these data sets are listed in Table 4.1 while additional information about them is given in the following subsections.

Table 4.1 Data sets used in our experimental work

| Data sets | Type of the data | # attributes | # patterns | # classes |
|---|---|---|---|---|
| Breast W | continuous | 9 | 683 | 2 |
| Diabetes | continuous | 8 | 768 | 2 |
| Glass | continuous | 9 | 214 | 6 |
| Heart C | continuous | 13 | 297 | 5 |
| Sonar | continuous | 60 | 208 | 2 |
| Wine | continuous | 13 | 178 | 3 |

### 4.4.1 Breast W

This data set, which is known as "Wisconsin breast cancer data"[13], was collected and analyzed by Wolberg, Street and Mangasarian (Street, Wolberg, & Mangasarian, 1993) from University of Wisconsin Hospitals. It consists of 699 cases with 9 attributes representing some properties of the cell, such as thickness, size and shape. The values of the features were obtained after assessed nuclear features of fine needle aspirates from patients whose 458 were diagnosed as benign cases and the rest, 241 cases, as malignant.

### 4.4.2 Diabetes (Pima Indian Diabetes Database)

This data which is known as Pima data set[14] was held by the National Institutes of Diabetes and Digestive and Kidney Diseases and donated by Vincent Sugulito from The Johns Hopkins University. It consists of 768 cases collected from a population living in Arizona, USA. Pima data contains 8 attributes and two classes, namely, the patients tested positive for diabetes and those with a negative test.

[13] Breast W data set can be downloaded from UCI repository at https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/ or from KEEL website at http://sci2s.ugr.es/keel/dataset.php?cod=73

[14] Pima data set can be downloaded from UCI repository at http://archive.ics.uci.edu/ml/machine-learning-databases/pima-indians-diabetes/ or from KEEL at http://sci2s.ugr.es/keel/dataset.php?cod=21

### 4.4.3 Glass

Glass dataset[15] was created by B. German from Forensic Science Service, UK and donated by Vina Spiehler, Ph.D., DABFT. The classification of the glass, which supposes to be left at the scene crime, can be used as evidence for criminological investigation if it is correctly identified. The data set includes 214 instances with nine attributes and six outputs[16]. The data set has been used by many researchers to evaluate their classification algorithms.

### 4.4.4 Heart C (Cleveland)

Heart C data set[17] used in this study was supplied by Robert Detrano, M.D., Ph.D. of the V.A. Medical Center, Long Beach, CA (Detrano et al., 1989). The original data has 76 attributes, but most of the studies use only 13 attributes that represent the clinical and non-invasive test results of 303 patients undergoing angiography. The total number of cases considered is 297 –after removing the cases with missing values-, out of which 160 are identified as patients without heart disease (HD) (class with value 0) while the other 137 cases are diagnosed as patients with heart disease (classes with values 1, 2, 3 and 4).

### 4.4.5 Sonar

Sonar data set[18] contains 208 instances with 60 attributes and two classes. The attribute values are sonar signals obtained by bouncing off a metal cylinder and a roughly cylindrical rock at various angles. These angles spanned 90 degrees for the metal and



---

[15] Glass data can be downloaded from UCI at http://archive.ics.uci.edu/ml/datasets/Glass+Identification Or from KEEL at http://sci2s.ugr.es/keel/dataset.php?cod=20

[16] Some repositories such as UCI (see at http://archive.ics.uci.edu/ml/machine-learning-databases/glass/glass.names) report that Glass data contains seven inputs but in fact the input number 4 is missing.

[17] Heart C data set can be downloaded from UCI repository at http://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/ or from KEEL website at http://sci2s.ugr.es/keel/dataset.php?cod=57

[18] Sonar data set can be downloaded from UCI repository at http://archive.ics.uci.edu/ml/machine-learning-databases/undocumented/connectionist-bench/sonar/ or from KEEL website at http://sci2s.ugr.es/keel/dataset.php?cod=85

180 degrees for the rock. The task of the classifier is to identify the sonar signal as either “Rock” or “Metal”.

### 4.4.6 Wine

Wine data set[19] was obtained from a chemical analysis of wines grown in Italy. It includes 178 instances and 13 features which represent the constituents found in each of the three types of wines (three classes).

## 4.5 Experimental setups

All the algorithms and methods were implemented using MATLAB R2013a. In addition, the five classifiers except SVMs can be found in Statistical and Neural Networks toolboxes of MATLAB R2013a. For SVM classifier, we used a well-known library implementation that can be downloaded from http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

To estimate the testing error rates for fuzzy rule-based systems and other classifiers, we followed the same method applied by (Ishibuchi & Nojima, 2007) in order to make a fair comparison with their results. For each data set, we run 10 independent iterations with different data partitions of 10-cross validation procedure (10×10cv). Then we calculate the average error rates over the 100 runs for each data.

The seven feature selection methods, used to rank the attributes and then to calculate the probability of their selection, are implemented in FEAST toolbox and can be downloaded from this link https://github.com/Craigacp/FEAST/. In addition, we downloaded random forest algorithm from https://code.google.com/p/randomforest-matlab/.

In Table 4.2, we list the parameter values used in our implementation. For other algorithms which are not listed below in Table 4.2 like statistical classifiers, we mention

[19] Wine data set can be downloaded from UCI repository at https://archive.ics.uci.edu/ml/machine-learning-databases/wine/ or from KEEL website at http://sci2s.ugr.es/keel/dataset.php?cod=31

that we used the default parameter values provided by MATLAB R2013a or other toolboxes.

Table 4.2 Parameter specifications for the algorithms used in the experiments

| **Phase** | **method** | **Parameter specifications** |
|---|---|---|
| Phase1: fuzzy rule-based system | Original | - Number of fuzzy sets per variable: 14 + “don’t care”.<br>- Reasoning method: single winner rule.<br>- Number of fuzzy rules generated for each fuzzy system in initial population: 20.<br>- Probability of “don’t care” is 0.95 for Sonar data and 0.80 for other data sets.<br>- Population size: 200.<br>- Crossover probability in the main part $P_c = 0.9$.<br>- Crossover probability in the Michigan-style part: 0.9.<br>- Mutation probability in the main part: $\frac{1}{n}$, where $n$ is the number of attributes.<br>- Mutation probability in the Michigan-style part: $\frac{1}{n}$.<br>-Number of generations: 5000.<br>- Multi-objective genetic algorithm: NSGA-II. |
| | Proposal1 | - All the parameter values for Proposal1 are the same as Original. In the following are the differences:<br>- Multi-objective genetic algorithm: Controlled Elitism NSGA-II instead of NSGA-II used in Original.<br>- The Pareto fraction (fraction of elite members in the population):0.35. |
| | Proposal2 | - All the parameter values for Proposal2 are the same as Original. In the following are the differences:<br>- Multi-objective genetic algorithm: Controlled Elitism NSGA-II.<br>- The Pareto fraction (fraction of elite members in the population):0.35.<br>- All the seven feature selection methods are used by their default parameters as they were implemented in FEAST toolbox without any change.<br>- Factor values in Figure 4.6 are as follows:<br>Factor=1 for Wine, Glass and<br>Factor=2 for Breast W, Heart C, Diabetes<br>Factor =36 for Sonar. |

| Phase | method | Parameter specifications |
|---|---|---|
| Phase2: Ensemble method | | - Random Forest (RF): number of members: 50 (RF is used as a benchmark method for comparison purpose).<br>- The number of base classifiers:05<br>- The number of ensemble members for each of the five classifiers: 50.<br>- We used the default parameters of all five classifiers as they are implemented in MATLAB R2013a.<br>**ANNs**<br>Number of neurons in input layer equals to the number of attributes.<br>Number of neurons in output layer equals to the number of classes.<br>Number of neurons in hidden layer $N_h$ is calculated as:<br>$N_h$ =( Number of attributes + number of classes)/2<br>***GA-based output Selection***<br>Controlled Elitism NSGA-II was applied with the following parameters:<br>- The Pareto fraction (fraction of elite members in the population):0.35.<br>- Population size: 10<br>- Number of generations: 200<br>- Probability of crossover: 1 |

## 4.6 Summary

We introduced in this chapter the proposed method that aims to combine the interpretability of the fuzzy rule-based system and the accuracy of the ensemble method in one system that can be used for both classification and data analysis. The objective of the first phase is to improve an existing fuzzy rule-based system that used NSGA-II to optimize both the accuracy and interpretability in the fuzzy system by proposing two variant methods. The first one, we named as Proposal1, in which we employed an enhanced version of NSGA-II called Controlled Elitism NSGA-II for accuracy-interpretability trade-off optimization. In the second variant method, which we named Proposal2, we improved the selection method of the antecedent conditions of the fuzzy systems generated in the initial population of GA algorithm. Unlike the method used in the Original algorithm, which uses a random selection of the antecedent conditions, we used feature-based selection method to favor the important features. In phase2, we proposed a design of an ensemble method that combines five different base classifiers and then apply GA-selection method to select a subset from all the ensemble outputs using accuracy and diversity measures as two objectives in the fitness function. In the last phase, we combine the two classifiers, namely, the fuzzy rule-based system and the ensemble method in one classification method. In the combined fuzzy-ensemble method, the fuzzy rule-based system is used for both classification and interpretation while the ensemble classifier is used only in the uncertainty, rejected and uncovered classifications. We introduced two methods, namely, Method1 and Method2 to estimate the threshold value under which the fuzzy rule-based system's classification is considered "uncertain" and thus rejected. In this case, as previously stated, the ensemble method is used for classification.

# CHAPTER5: RESULTS AND DISCUSSIONS

In this chapter, we present the results achieved by the proposed method and conduct various comparisons with benchmark methods. The first part of this chapter details the results of the improved fuzzy rule-based system and compares it with the original version of the algorithm that was proposed in (Ishibuchi & Nojima, 2007). This algorithm, as I mentioned in Chapter 4, is the result of a series of enhancements published in different articles (Ishibuchi et al., 2001; Ishibuchi & Yamamoto, 2004; Ishibuchi, Yamamoto, et al., 2005) and it is frequently used as a benchmark method for result comparisons (Y. Nojima et al., 2014; Pulkkinen & Koivisto, 2008). In the second part, which is related to ensemble methods, the classification results produced from different ensemble methods are discussed and analyzed. Finally, the performance of the fuzzy-ensemble classifier is presented in the last part of this chapter.

## 5.1 Fuzzy rule-based system

### 5.1.1 Comparison between Original, Proposal1 and Proposal2

As we explained in the previous chapter, the objective of introducing two variants of the original algorithm proposed in (Ishibuchi & Nojima, 2007) is to improve the trade-off between the accuracy and the interpretability of the fuzzy rule-based system. Tables from 5.1 to 5.6 show the non-dominated fuzzy rule-based systems obtained by the Original method and the two variant methods for Breast W and Wine data sets. To avoid putting lot of tables for the sake of readability, the tables that show results of the other four data sets are placed in Appendix-A (from Tables A.1 to A.12). The results of the non-dominated fuzzy rule-based systems obtained by the three proposals represent the interpretability, expressed in terms of number of rules and the total rule length, and also classification accuracy using the testing error rate.

To make the results more understandable, we plot the results listed in the tables into figures. For example, Table 5.1 to 5.3 related to Breast W data set, are plotted into Figure 5.1. In addition, it is worthy to mention that we use “total rule length” in the $x$-axis rather than “number of rules” to avoid ambiguity in the plotted graphs as there is more than one solution that have the same number of rules. For example, in Table 5.1, we have 4 solutions that each has three rules. Furthermore, figures display only the solutions that achieved the best accuracy. For example, in Figure 5.1, we avoided displaying solutions that have 1 or 0 total rule length because their results are significantly lower than the rest of solutions and including them in the plot makes the marks that pinpoint to the solutions in the graph look small and highly overlap which make them less visible.

We can notice from the results of Wine data set which are shown in Tables 5.4 to 5.6 and Figure 5.2 that there is a clear dominance of Proposal2 over the other methods. But for other data sets, there is no algorithm which dominates all the solutions. For example, in Breast W data set whose results are shown in Tables 5.1 to 5.3 and Figure 5.1, we can notice from the five plotted solutions, which represent about half of all non-dominated solutions, that Proposal2 received the best error rate in 3 of them, against 2 for Proposal1.

For results comparison, we used the same method of evaluation applied in (Ishibuchi & Nojima, 2007) and (Pulkkinen & Koivisto, 2008). The former study is where the original algorithm was proposed while the latter study proposed a method for generating a fuzzy rule-based classifier. The evaluation method compares the algorithms based on how many solutions a given algorithm dominates the other algorithms. The best algorithm is the one that can get the highest number of solutions in which it dominates the other algorithms. For each data set, the solutions which are listed in Tables 5.1 to 5.6 and Tables A.1 to A.12 and have the same number of rules and antecedent conditions are compared and ranked according to the best error rate achieved in testing data set, then the average of rank is calculated. The best solution is given rank 1 while rank 2 is for the second best solution and the last one is ranked 3. In order for a solution to be considered in the comparison, two algorithms at least must have this solution with the same number of rules and antecedent conditions. Tables 5.7 and 5.8 summarize the results and provide average ranks of the three methods based on testing error rates. We can notice that proposal2 achieved the best average ranks (1.58) followed by Proposal1 as second best average rank (1.75) then Original (2.67). More specifically, Proposal2 achieved the best rank in three data sets, namely, Diabetes, Heart C and Wine while shared the best rank with Proposal1 in Sonar data set. For Proposal1, it achieved the best rank in two data sets: Breast W and Glass. The Original method achieved the second best average rank in Glass and Heart data sets.

For the best testing error, we can see from Table 5.9 that Proposal2 has consistently achieved the best results for all the six data sets which shows that Proposal2 has successfully found high-quality solutions.

For Proposal2, Table 5.6 and its corresponding Figure 5.2 show that the fuzzy rule-based system which has 3 rules and 5 antecedent conditions achieved a better result than that of 3 rules and 6 fuzzy antecedent conditions. This behavior was repeated also in other data sets such as Diabetes in Table A.3, Heart C data in Table A.9 and Sonar data in Table A.12. We can say that, unlike training data set, there is no consistency in the accuracy-interpretability trade-off in testing data sets. The same conclusion is also correct for Original and Proposal1.

*Number of non-dominated solutions generated for each method*

For the results of Breast W data set shown in Tables 5.1 to 5.3, we notice that Proposal1 and 2 have one extra solution (#rules=4, Total rule length=8) which the Original algorithm does not have. It is likely, in practical situations, that not all the solutions are important for the user as he/she will generally choose one or few solutions based on his/her preferences but having more solutions is an advantage for the algorithm, as it allows for more choices. Table 5.10 shows the average number of obtained non-dominated fuzzy rule-based systems for each proposal and the average ranks in the six data sets. We notice that Proposal2 achieved the best average rank (1.17) followed by Proposal1 (1.83) then Original (3). In addition, we can see that the average number of non-dominated solutions for Proposal1 and 2 are close to each other but they are significantly higher compared to Original.

One may wonder why the number of non-dominated solutions for the three methods reported in Tables 5.1 to 5.6 and Tables A.1 to A.12 are less than the number reported in Table 5.10. The answer is that the solutions listed in Tables 5.1 to 5.6 and A.1 to A.12 do not represent all the solutions found in the 100 runs. In fact, they represent only the solutions with a given number of rules and antecedent conditions that are present in at least 51 out of the 100 runs. But what is reported in Table 5.10 are all the solutions including those which are present in 50 or less out of the 100 runs.

Table 5.1 Non-dominated fuzzy rule-based systems generated from Breast W data set using Original method ((Ishibuchi & Nojima, 2007))

| # of rules | Total rule length | Error rate on training patterns | Error rate on test patterns |
|---|---|---|---|
| 1 | 0 | 34.99 | 34.99 |
| 2 | 1 | 9.69 | 11.67 |
| 2 | 2 | 5.23 | 6.73 |
| 2 | 3 | 3.48 | 4.25 |
| 2 | 4 | 2.94 | 3.88 |
| 2 | 5 | 2.80 | 3.51 |
| 3 | 2 | 4.44 | 5.02 |
| 3 | 5 | 2.49 | 4.32 |
| 3 | 6 | 2.36 | 4.09 |
| 3 | 7 | 2.24 | 4.07 |

Table 5.2 Non-dominated fuzzy rule-based systems generated from Breast W data set using Proposal1

| # of rules | Total rule length | Error rate on training patterns | Error rate on test patterns |
|---|---|---|---|
| 1 | 1 | 34.99 | 35.18 |
| 2 | 1 | 9.74 | 11.81 |
| 2 | 2 | 5.30 | 5.61 |
| 2 | 3 | 3.36 | 3.75 |
| 2 | 4 | 2.87 | 3.89 |
| 2 | 5 | 2.72 | 3.67 |
| 3 | 2 | 4.44 | 4.58 |
| 3 | 5 | 2.39 | 4.18 |
| 3 | 6 | 2.22 | 4.22 |
| 3 | 7 | 2.11 | 3.92 |
| 4 | 8 | 1.80 | 3.52 |

Table 5.3 Non-dominated fuzzy rule-based systems generated from Breast W data set using Proposal2

| # of rules | Total rule length | Error rate on training patterns | Error rate on test patterns |
|---|---|---|---|
| 1 | 1 | 34.98 | 35.15 |
| 2 | 1 | 9.72 | 11.59 |
| 2 | 2 | 5.30 | 6.32 |
| 2 | 3 | 3.38 | 3.92 |
| 2 | 4 | 2.86 | 4.08 |
| 2 | 5 | 2.68 | 4.45 |
| 3 | 2 | 4.39 | 5.14 |
| 3 | 5 | 2.46 | 3.83 |
| 3 | 6 | 2.28 | 3.79 |
| 3 | 7 | 2.08 | 4.26 |
| 4 | 8 | 1.91 | 3.37 |

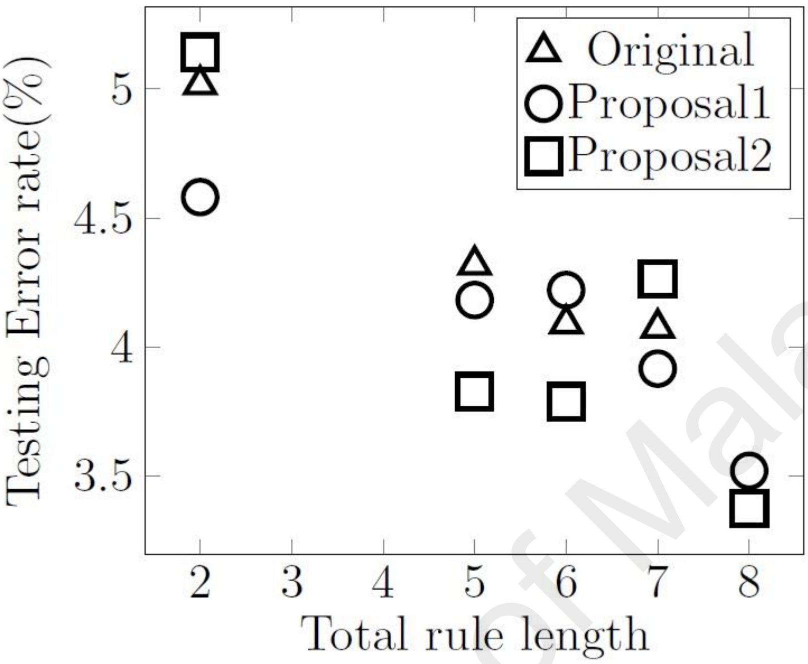


Figure 5.1 Results of testing error rates on Breast W data set for Original, Proposal1 and Proposal2

Table 5.4 Non-dominated fuzzy rule-based systems generated from Wine data set using Original method ((Ishibuchi & Nojima, 2007))

| # of rules | Total rule length | Error rate on training patterns | Error rate on test patterns |
|---|---|---|---|
| 1 | 1 | 69.36 | 70.06 |
| 2 | 2 | 30.48 | 33.25 |
| 2 | 3 | 28.55 | 31.12 |
| 2 | 4 | 27.71 | 30.63 |
| 3 | 3 | 8.40 | 13.45 |
| 3 | 4 | 5.35 | 10.34 |
| 3 | 5 | 3.53 | 8.32 |
| 3 | 6 | 2.24 | 6.90 |
| 3 | 7 | 1.43 | 5.35 |

Table 5.5 Non-dominated fuzzy rule-based systems generated from Wine data set using Proposal1

| # of rules | Total rule length | Error rate on training patterns | Error rate on test patterns |
|---|---|---|---|
| 1 | 1 | 60.11 | 61.11 |
| 2 | 2 | 30.70 | 32.72 |
| 2 | 3 | 28.11 | 29.96 |
| 2 | 4 | 27.50 | 29.89 |
| 3 | 3 | 7.48 | 11.49 |
| 3 | 4 | 4.70 | 9.72 |
| 3 | 5 | 2.37 | 6.87 |
| 3 | 6 | 1.44 | 5.53 |
| 3 | 7 | 0.91 | 5.52 |

Table 5.6 Non-dominated fuzzy rule-based systems generated from Wine data set using Proposal2

| # of rules | Total rule length | Error rate on training patterns | Error rate on test patterns |
|---|---|---|---|
| 1 | 1 | 60.08 | 61.51 |
| 2 | 2 | 30.84 | 32.00 |
| 2 | 3 | 28.39 | 29.73 |
| 2 | 4 | 27.63 | 30.15 |
| 3 | 3 | 7.57 | 10.99 |
| 3 | 4 | 4.76 | 8.18 |
| 3 | 5 | 2.59 | 5.15 |
| 3 | 6 | 1.50 | 5.28 |
| 3 | 7 | 0.92 | 4.23 |

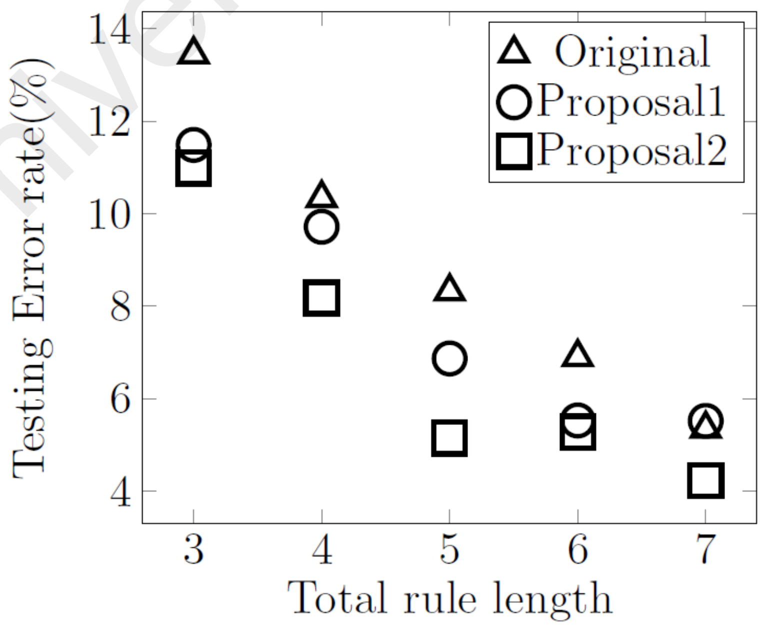


Figure 5.2 Results of testing error rates on Wine data set for Original, Proposal1 and Proposal2

Table 5.7 Average ranks of testing error rates for Original, Proposal1 and Proposal2

| | Original | Proposal1 | Proposal2 |
|---|---|---|---|
| Breast W | 2.11 | **1.8** | 2 |
| Diabetes | 2.1 | 1.86 | **1.71** |
| Glass | 1.88 | **1.48** | 1.95 |
| Heart C | 1.83 | 2.12 | **1.63** |
| Sonar | 2.3 | **1.77** | **1.77** |
| Wine | 2.89 | 1.89 | **1.22** |

Table 5.8 Sort of average ranks reported in Table 5.7 for Original, Proposal1 and Proposal2

| | Original | Proposal 1 | Proposal 2 |
|---|---|---|---|
| Breast W | 3 | 1 | 2 |
| Diabetes | 3 | 2 | 1 |
| Glass | 2 | 1 | 3 |
| Heart C | 2 | 3 | 1 |
| Sonar | 3 | 1.5 | 1.5 |
| Wine | 3 | 2 | 1 |
| Average ranks | 2.67 | 1.75 | **1.58** |

Table 5.9 Average ranks of best testing error rates for Original, Proposal1 and Proposal2 extracted from Tables 5.1 to 5.6 and A.1 to A.12

| | | # of rules | Total rule length | Error rate on test patterns | Rank |
|---|---|---|---|---|---|
| Breast W | Original | 3 | 7 | 4.07 | 3 |
| | Proposal1 | 4 | 8 | 3.52 | 2 |
| | Proposal2 | 4 | 8 | **3.37** | **1** |
| Diabetes | Original | 3 | 8 | 24.68 | 2 |
| | Proposal1 | 5 | 14 | 24.84 | 3 |
| | Proposal2 | 5 | 14 | **24.01** | **1** |
| Glass | Original | 4 | 14 | 40.59 | 3 |
| | Proposal1 | 5 | 12 | 39.08 | 2 |
| | Proposal2 | 5 | 12 | **37.30** | **1** |
| Heart C | Original | 4 | 11 | 46.33 | 2 |
| | Proposal1 | 5 | 16 | 46.62 | 3 |
| | Proposal2 | 5 | 17 | **45.43** | **1** |
| Sonar | Original | 3 | 8 | 24.34 | 3 |
| | Proposal1 | 4 | 10 | 24.03 | 2 |
| | Proposal2 | 4 | 13 | **22.94** | **1** |
| Wine | Original | 3 | 7 | 5.35 | 2 |
| | Proposal1 | 3 | 7 | 5.52 | 3 |
| | Proposal2 | 3 | 7 | **4.23** | **1** |
| Average rank | Original | | | | 2.5 |
| | Proposal1 | | | | 2.5 |
| | Proposal2 | | | | **1** |

Table 5.10 Average number of obtained non-dominated fuzzy rule-based systems for Original, Proposal1 and Proposal2

| | Original | Proposal 1 | Proposal 2 |
|---|---|---|---|
| Breast W | 12.25 | 18.21 | 18.20 |
| Diabetes | 17.06 | 29.98 | 30.58 |
| Glass | 27.09 | 39.69 | 39.70 |
| Heart C | 18.59 | 37.97 | 36.36 |
| Sonar | 17.66 | 33.75 | 34.10 |
| Wine | 11.81 | 15.78 | 16.15 |
| Average ranks | 3.00 | 1.83 | **1.17** |

*Comparison of average best error rates on training patterns*

In this comparison, we select, among the non-dominated solutions, the fuzzy rule-based system with the best error rate regardless the number of the rules and the antecedent conditions. The average error rate is calculated over the 100 runs for each method.

Table 5.11 displays the average best error rates on training sets. The results on the table show that Proposal2 received the best average rank (1.25) then Proposal1 which achieved an average rank equal to 1.75 followed by the Original algorithm with average rank 3. For the testing rates that correspond to the best error rates on training sets, we notice from Table 5.12, that Proposal2 performed better than Proposal1 achieving average rank equal to 1.17 compared to 1.83 for Proposal1. The results of average testing error for the Original algorithm are not available for comparison. So, from these results, we can observe that Proposal2 achieved better results than Proposal1 for both training and testing data sets which reflects its search and generalization ability over Proposal1. In addition, from the results of Tables 5.7 to 5.9, we can say that Proposal1 and 2 have generally better search and generalization ability compared to Original. This indicates the potential of controlled NSGA-II algorithm compared to NSGA-II.

Table 5.11 Average best error rates (in %) on training patterns among the obtained fuzzy rule-based systems

| | Original | Proposal1 | Proposal2 |
|---|---|---|---|
| | training | training | training |
| Breast W | 1.74 | **1.51** | 1.53 |
| Diabetes | 19.59 | 18.43 | **18.32** |
| Glass | 25.94 | 25.90 | **25.61** |
| Heart C | 34.59 | 32.49 | **31.34** |
| Sonar | 8.42 | 5.48 | **5.36** |
| Wine | 0.03 | **0** | **0** |
| Average ranks | 3 | 1.75 | 1.25 |

Table 5.12 Average test error rates (in %) of fuzzy rule-based systems that achieved best error rates on training patterns among the obtained fuzzy rule-based systems

| | Proposal1 | Proposal2 |
|---|---|---|
| | testing | testing |
| Breast W | 3.94 | 3.85 |
| Diabetes | 24.24 | 24.56 |
| Glass | 38.07 | 38.01 |
| Heart C | 47.41 | 46.06 |
| Sonar | 23.40 | 22.95 |
| Wine | 5.46 | 5.41 |
| Average ranks | **1.83** | **1.17** |

### 5.1.2 Selection of a suitable fuzzy rule-based system among non-dominated fuzzy systems

As we previously stated, in real situations, the user has to select a solution based on his/her preferences among non-dominated solutions. For our case, we prefer to select the fuzzy rule-based systems that have the best classification ability which are generally, but not consistently, the fuzzy systems with the highest number of rules and antecedent conditions (i.e. the least interpretable). To adopt consistent criteria, we choose the fuzzy rule-based system with the highest number of rules as it is likely to be the most accurate solution.

The selection process of these fuzzy systems are as the following. Since, for each data set, we have run 100 training and testing experiments (10 iterations of ten-cross-validation procedure or simply 10×10cv), all the solutions for training data are sorted in descending order by two criteria: firstly by the number of rules and subsequently, in case of tie, the number of antecedent conditions. Then one solution or one fuzzy rule-based system, which comes at the top of the sorted solutions, is selected for each training data and after that used to calculate the corresponding testing accuracy. Finally, the average error rate is calculated by averaging the error rate values of the 100 runs. Table 5.13 shows the results of both Proposal1 and 2 which is quite similar to the results of Tables 5.11 and 5.12. The difference is due to the fact that it is not necessarily the fuzzy rule-based system which has the highest number of rules and antecedent conditions is the one which has the highest training accuracy. Table 5.13 confirms the previous results that Proposal2 achieved better results than Proposal1 for both training and testing data set. This indicates that using feature selection-based approach to select the “don’t care” antecedent conditions in the initial population may give better results compared to the use of random selection method used in Proposal1 and Original. Actually, this conclusion is consistent with previous findings which state that the start from a good initial population may help Genetic algorithm to find better solutions than if it starts from a randomly generated initial population (Grosan & Abraham, 2007).

Since we have to select one fuzzy rule-based system in order to combine it with the ensemble method, so we chose Proposal2.

Table 5.13 Average error rates of the selected fuzzy rule-based systems

| | Proposal1 | | Proposal2 | |
|---|---|---|---|---|
| | training | testing | training | testing |
| Breast W | 1.52 | 3.96 | 1.54 | 3.82 |
| Diabetes | 18.43 | 24.29 | 18.33 | 24.56 |
| Glass | 25.91 | 37.99 | 25.62 | 38.10 |
| Heart C | 32.50 | 47.41 | 31.35 | 46.03 |
| Sonar | 5.48 | 23.35 | 5.36 | 22.85 |
| Wine | 0.50 | 5.27 | 0.44 | 5.03 |
| Average ranks | 1.83 | 1.67 | **1.17** | **1.33** |

### 5.1.3 Comparison between the selected fuzzy rule-based system and benchmark methods

One question may arise that selecting less interpretable fuzzy systems in terms of complexity can be a disadvantage for our solution. But our choice is justified by the observation that even the least interpretable fuzzy rule-based systems obtained from Proposal2 have modest numbers of rules and antecedent conditions.

To evaluate our choice, we make a comparison between the fuzzy rule-based systems obtained by proposal2 that have the highest number of rules and antecedent conditions and the commonly used fuzzy rule-based systems proposed in the literature. The comparison is made based on two criteria, namely, accuracy and interpretability expressed in terms of the number of rules and number of antecedent conditions per rule. The results listed in Tables 5.14 and 5.15 about the algorithms FURIA, SLAVE and CHI are obtained from (Hühn & Hüllermeier, 2009). In the aforementioned article, the authors proposed a fuzzy rule-based classifier called FURIA and they include SLAVE and CHI algorithms as benchmark methods for comparison. They estimated the error rates as the following: the data set was randomly split into 2/3 for training and 1/3 for testing. This process is repeated 100 times to stabilize the results. The results on Heart C data set were not reported because the authors used a different version of the data set that has two classes instead of five classes as in our study. The other algorithm, we named Hybrid, is proposed in (Ishibuchi, Yamamoto, et al., 2005) and the results reported are calculated using the same method applied in our study, i.e. 10×10cv. In fact, Hybrid algorithm is a variant and early version of the Original algorithm. From Table 5.12, which summarizes the testing error rates, we notice that Proposal2 received the best rank with 1.67 followed by Hybrid with 1.83 then FURIA with 2.40.

If we take only Proposal2 and Hybrid and compare them in term of accuracy, we can see that Proposal2 outperformed Hybrid in 3 data sets (Diabetes, Heart C and Sonar) while it is inferior on the other 3 data sets (Breast W, Glass and Wine). Thus, we can say they are equal in term of accuracy. But as we can see from Table 5.15, Proposal2 has fewer rules than Hybrid in 5 out of 6 data sets, which indicates that, by taking accuracy and interpretability measures into consideration, Proposal2 achieved in overall better results than Hybrid.

For FURIA algorithm, Table 5.14 shows that Proposal2 achieved better error rates in 4 out of 5 data sets which indicates its classification ability compared to FURIA. For interpretability measure, Proposal2 and FURIA have comparable performance. Proposal2 has fewer rules in 3 out of 5 data sets while FURIA has shorter rules in 4 out of 5 data sets. As a result, we can conclude that for both interpretability and accuracy measures, Proposal2 performed better than FURIA.

In the case of SLAVE algorithm, Proposal2 outperformed SLAVE in term of accuracy in 5 out of 5 data sets. For interpretability measure, both of the algorithms have comparable performance. SLAVE has fewer rules in 3 out of 5 data sets while Proposal2 has shorter rules in 4 out of 5 data sets. In overall results, Proposal2 received better results than SLAVE.

For CHI algorithm, because of its lacks of a mechanism for reducing the number of rules and antecedent conditions, we can notice that it has a high number of rules compared to other methods. In addition to its complexity, CHI achieved less accuracy than Proposal2 in all the 5 data sets.

As we can see from this comparison with benchmark methods, Proposal2 is competitive or even better than these methods in both accuracy and interpretability measures. So, our approach to select the most complicated fuzzy rule-based systems seems to be an acceptable approach for our case.

Table 5.14 Average testing error rates for Proposal2 and some benchmark methods

| | Proposal2 | FURIA | SLAVE | Ishi | CHI |
|---|---|---|---|---|---|
| Breast W | 3.82 | 4.32 | 4.51 | **3.54** | 9.8 |
| Diabetes | **24.56** | 25.29 | 26.35 | 25.08 | 27.45 |
| Glass | 38.10 | **31.78** | 38.17 | 37.80 | 38.61 |
| Heart C | **46.03** | / | / | 46.50 | / |
| Sonar | **22.85** | 22.99 | 31.5 | 23.70 | 25.39 |
| Wine | 5.03 | 6.75 | 7.54 | **4.94** | 7.23 |
| Average rank | **1.67** | 2.40 | 4.40 | 1.83 | 4.60 |

Table 5.15 Average number of rules and antecedent conditions per rule for Proposal2 and some benchmark methods

| | Proposal 2 | | FURIA | | Ishi | | SLAVE | | CHI | |
|---|---|---|---|---|---|---|---|---|---|---|
| | #Rule | #condi | #Rule | #condi | #Rule | #condi | #Rule | #condi | #Rule | #condi |
| Breast W | **5.2** | 2.7 | 12.2 | 2.9 | 10 | / | 5.8 | 3.7 | 172.4 | / |
| Diabetes | 10.44 | 3.48 | **8.5** | 2.6 | 10 | / | 9.3 | 3.7 | 98.6 | / |
| Glass | **7.61** | 3.51 | 11.3 | 2.2 | 10 | / | 12.3 | 3.3 | 42.7 | / |
| Heart C | **8.77** | 4.09 | / | / | 10 | / | / | / | / | / |
| Sonar | 8.78 | 4.36 | 8.1 | 2.3 | 10 | / | **6.9** | 4.7 | 137.1 | / |
| Wine | 4.07 | 2.11 | 6.2 | 1.9 | 10 | / | **3.8** | 2.9 | 101.2 | / |
| Average rank | **2** | 2 | 2.6 | **1.2** | 3 | / | **2** | 2.8 | 5 | / |

## 5.2 Ensemble methods

In this section, we conduct a series of comparisons to determine which of the ensemble methods is more accurate in order to be selected for the fuzzy-ensemble method. Some comparisons also aim at analyzing classifiers' performances as single and ensemble classifiers.

### 5.2.1 Comparison between single and ensemble classifiers

By examining Table 5.21 and the corresponding figures -from Figures 5.3 to 5.7-, which summarize the results presented in Tables 5.16 to 5.20, we notice that single classifiers did not get the best rank in all the five classifiers which indicates that their performances were increased by using them as base classifiers in the ensemble methods. As an example, we show in Figure 5.8 how RLO ensemble methods have contributed to the improvement of testing accuracy rates of the five single classifiers applied on sonar data set.

In fact, by studying figures from 5.9 to 5.14, we notice that the two single classifiers that benefited the most from the ensemble methods are CART which ranks 6 and ANNs with rank 5.33. In addition, from Table 5.22 and Figure 5.9, we can see that both of the two classifiers have made clearly significant gains of accuracy rate (more than 7%) compared with other classifiers (less than 2% for NB and less than 1% for LDA and SVM). These results are also reflected in Tables 5.23 and 5.24 and especially Table 5.25 and Figure 5.15 where we can notice a significant improvement of ANNs whose rank as a single classifier was 3.33 (the fourth out of 5) and then becomes, as base classifier, at the top of all ensemble methods. CART also enhanced its rank from 4.33 ($5^{th}$ place) as a single classifier to 3.8 ($4^{th}$ place) as a base classifier.

This result confirms previous findings which stated that CART and ANNs are unstable classifiers and thus are suitable for ensemble methods (Ludmila I Kuncheva, 2014).

For ANNs, the instability is due to the randomness of the weight initialization in the training phase while CART is unstable because small changes in the training sets can produce very different trained classifiers (Ludmila I Kuncheva, 2014). This behaviour has made decision tree algorithms, such CART and C4.5, the favourite base classifier for many proposed ensemble methods such Bagging (L. Breiman, 1996) and AdaBoost (Freund & Schapire, 1997).

In addition to ANNs and CART, we can see from Figure 5.12 that NB as base classifier gained accuracy but it was modest compared to ANNs and CART. This made NB, as it is shown in Table 5.25, ranks the last among the ensemble methods compared to 3rd place as a single classifier.

For LDA and SVM classifiers which their results are depicted in Figures 5.10 and 5.14, respectively; they gained in Bagging modest improvements (around 0.2%) and relatively higher in RLO and RSO (between 1.2% and 1.8% for LDA and 2.2% for SVMs). This indicates that SVMs and LDA are more stable than the other classifiers.

As a base classifier, Table 5.25 and Figure 5.15 show that SVMs have overall achieved the second best rank in ensemble methods after ANNs compared to the best rank as a single classifier. This drop in rank is due to the modest gains in accuracy compared to ANNs.

In addition, LDA and CART base classifiers achieved the second best rank in RS and AdaBoost, respectively. In overall, because LDA gained modest accuracy improvements, its rank dropped from 2.33 as a single classifier (2nd place) to 3 (3rd place) as a base classifier for the ensemble methods.

### 5.2.2 Comparison between the ensemble methods Bagging, Random Subspace, RLO, RSO and AdaBoost

In order to make comparisons between the ensemble methods, namely, Bagging, Random Subspace, RLO, RSO and AdaBoost and to know which of these methods achieved the best results, we created Tables 5.26 and 5.27 in addition to Figure 5.16 which summarizes the results of the two tables. From the tables, we notice that RLO achieved the best results in three base classifiers: LDA, ANNs and SVMs while shared the best results with RSO in NB. In addition, Bagging achieved the best results for base classifier CART. In addition, in Table 5.22, we can notice that RLO, Bagging and RSO have successfully enhanced the accuracy of all single classifiers. Interestingly, RLO and RSO have averagely achieved better gains than Bagging especially for stable classifiers such LDA and SVMs. This may indicate that, as the authors of RLO and RSO claimed, that RLO (L. Kuncheva & J. J. Rodriguez, 2007) and RSO (Rodríguez & Kuncheva, 2007) are good methods for creating diversity in ensemble methods.

The best result that AdaBoost achieved is the second place in CART. In addition, it enhanced the accuracy of single classifiers in three cases: CART, NB and ANNs. For RS, the best result achieved by this ensemble method is the third place for ANNs and its accuracy is better than single classifier only in two cases: CART and ANNs. AdaBoost and RS were not able to increase the accuracy of both LDA and SVMs. Thus, using SVMs and LDA as single classifiers is better than used them in AdaBoost or RS.

Table 5.16 Average testing error rates (in %) for single LDA classifier and ensemble methods using LDA

| | S(LDA) | Bag(LDA) | RS(LDA) | RLO(LDA) | RSO(LDA) | ADA(LDA) |
|---|---|---|---|---|---|---|
| Breast W | 4.01 | 4 | 4.11 | 3.37 | 3.63 | 4.28 |
| Diabetes | 22.63 | 22.59 | 23.2 | 22.97 | 23.37 | 26.44 |
| Glass | 38.36 | 37.99 | 39.32 | 34.29 | 34.71 | 38.76 |
| Heart C | 41.12 | 40.95 | 41.32 | 41.75 | 41.97 | 42.27 |
| Sonar | 25.26 | 24.98 | 24.41 | 21.29 | 22.44 | 24.15 |
| Wine | 1.46 | 1.51 | 1.68 | 1.7 | 1.92 | 1.51 |

Table 5.17 Average testing error rates (in %) for single CART classifier and ensemble methods using CART

| | S(CART) | Bag(CART) | RS(CART) | RLO(CART) | RSO(CART) | ADA(CART) |
|---|---|---|---|---|---|---|
| Breast W | 5.08 | 3.51 | 3.67 | 3.53 | 3.57 | 3 |
| Diabetes | 29.31 | 23.87 | 26.71 | 24.85 | 25.27 | 24.2 |
| Glass | 31.39 | 25.12 | 25.75 | 26.71 | 26.96 | 30.3 |
| Heart C | 51.44 | 44.46 | 45.86 | 46.9 | 45.68 | 42.94 |
| Sonar | 28.36 | 19.43 | 20.93 | 15.68 | 14.07 | 20.5 |
| Wine | 9.86 | 3.67 | 6.93 | 5.58 | 6.31 | 4.35 |

Table 5.18 Average testing error rates (in %) for single Naïve Bayes classifier and ensemble methods using Naïve Bayes

| | S(NB) | Bag(NB) | RS(NB) | RLO(NB) | RSO(NB) | ADA(NB) |
|---|---|---|---|---|---|---|
| Breast W | 3.25 | 3.18 | 3.51 | 3.26 | 3.18 | 3.39 |
| Diabetes | 26.48 | 25.65 | 27.04 | 24.72 | 24.74 | 26.21 |
| Glass | 40.51 | 36.48 | 40.18 | 35.02 | 32.37 | 36.8 |
| Heart C | 44.15 | 44.28 | 43.35 | 43.61 | 43.74 | 42.81 |
| Sonar | 23.59 | 24.02 | 24.14 | 19.52 | 17.41 | 25.84 |
| Wine | 2.26 | 2.55 | 2.14 | 2.78 | 3.5 | 2.37 |

Table 5.19 Average testing error rates (in %) for single ANNs classifier and ensemble methods using ANNs

| | S(ANN) | Bag(ANN) | RS(ANN) | RLO(ANN) | RSO(ANN) | ADA(ANN) |
|---|---|---|---|---|---|---|
| Breast W | 3.53 | 2.99 | 2.97 | 2.97 | 2.99 | 3.09 |
| Diabetes | 25.84 | 22.9 | 23.15 | 22.93 | 23.66 | 25.21 |
| Glass | 50.47 | 36.38 | 37.48 | 34.97 | 36.11 | 35.75 |
| Heart C | 42.07 | 42.62 | 43.66 | 43.14 | 42.64 | 41.14 |
| Sonar | 22.86 | 16.44 | 17.03 | 13.74 | 13.95 | 19.29 |
| Wine | 6.07 | 1.91 | 1.46 | 2.24 | 2.07 | 1.69 |

Table 5.20 Average testing error rates (in %) for single SVMs classifier and ensemble methods using SVMs

| | S(SVM) | Bag(SVM) | RS(SVM) | RLO(SVM) | RSO(SVM) | ADA(SVM) |
|---|---|---|---|---|---|---|
| Breast W | 3.43 | 3.37 | 3.31 | 3.24 | 3.31 | 3.18 |
| Diabetes | 22.71 | 22.54 | 23.46 | 22.94 | 23.16 | 24.79 |
| Glass | 38.53 | 38.21 | 40.6 | 34.15 | 32.71 | 43.24 |
| Heart C | 40.91 | 39.95 | 41.73 | 41.36 | 42.5 | 41.64 |
| Sonar | 21.84 | 22.32 | 22.39 | 16.96 | 16.91 | 24.05 |
| Wine | 4.41 | 4.57 | 3.74 | 3.59 | 3.69 | 5.24 |

Table 5.21 Average ranks of testing error rates for single classifiers and their ensemble methods

| Algorithm | LDA | CART | NB | ANNs | SVMs |
|---|---|---|---|---|---|
| S | 3.17 | 6.00 | 4.00 | 5.33 | 3.50 |
| Bag | 2.58 | **1.67** | 3.58 | 2.92 | 3.17 |
| RS | 4.33 | 4.33 | 4.17 | 3.42 | 4.42 |
| RLO | **2.50** | 3.17 | 2.83 | **2.58** | **2.17** |
| RSO | 3.67 | 3.33 | **2.58** | 3.42 | 2.92 |
| ADA | 4.75 | 2.50 | 3.83 | 3.33 | 4.83 |

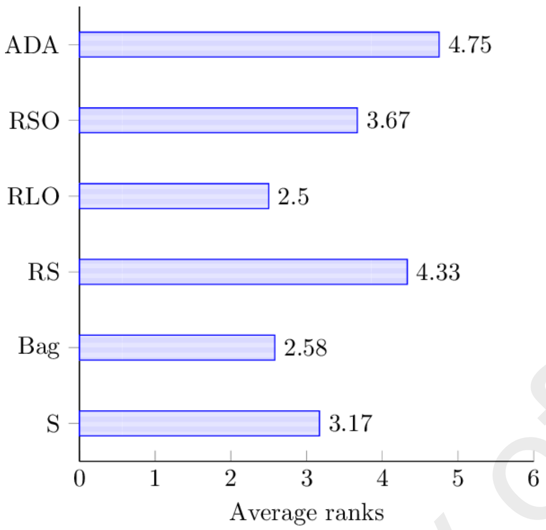


Figure 5.3 Average ranks of testing error rates for single LDA and ensemble methods using LDA

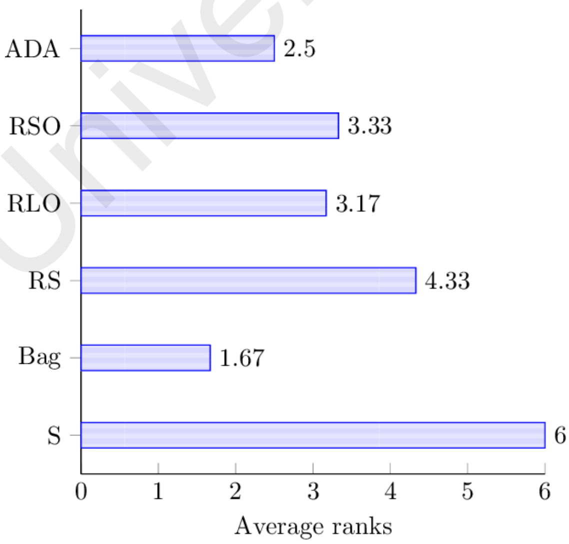


Figure 5.4 Average ranks of testing error rates for single CART and ensemble methods using CART

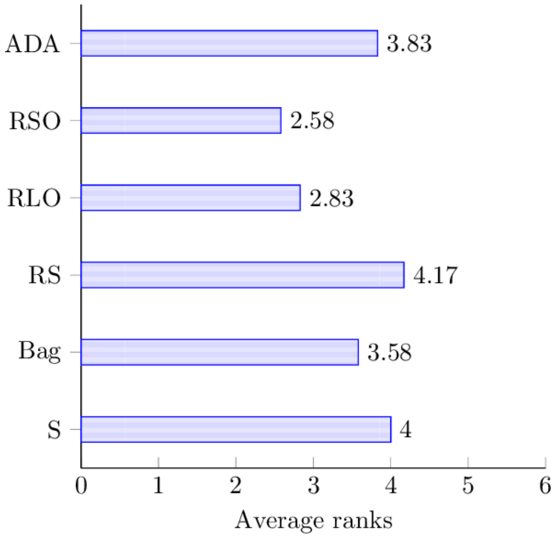


Figure 5.5 Average ranks of testing error rates for single NB and ensemble methods using NB

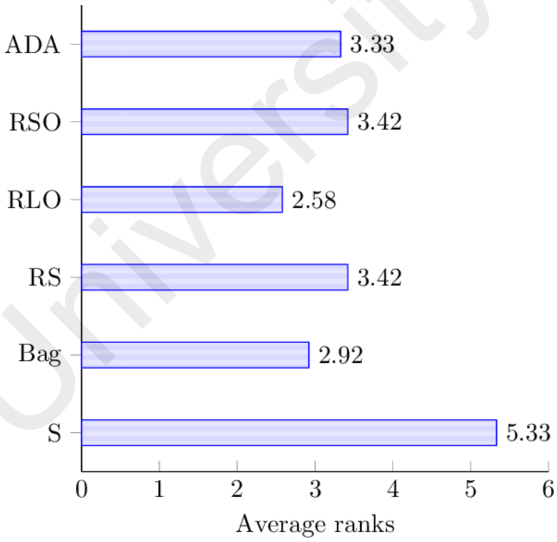


Figure 5.6 Average ranks of testing error rates for single ANNs and ensemble methods using ANNs

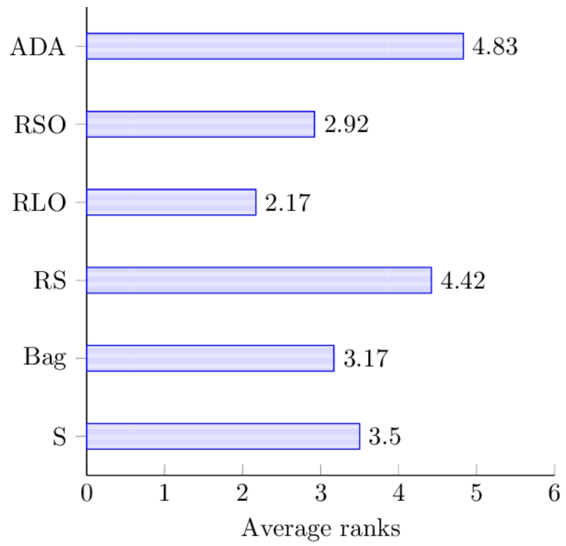


Figure 5.7 Average ranks of testing error rates for single SVMs and ensemble methods using SVMs

Table 5.22 Average percentage gains of accuracy (in %) for ensemble methods with respect to the single classifier (positive and negative means gain and loss of accuracy, respectively)

| Algorithm | Bag | RS | RLO | RSO | ADA | Average |
|---|---|---|---|---|---|---|
| **LDA** | 0.21 | -0.30 | 1.80 | 1.20 | -1.06 | 0.37 |
| **CART** | 8.70 | 6.42 | 7.76 | 8.25 | 7.60 | **7.746** |
| **NB** | 1.15 | 0.06 | 2.90 | 3.95 | 0.97 | 1.806 |
| **ANNs** | 7.46 | 6.69 | 8.31 | 7.89 | 6.99 | **7.468** |
| **SVMs** | 0.27 | -0.93 | 2.23 | 2.23 | -2.50 | 0.26 |

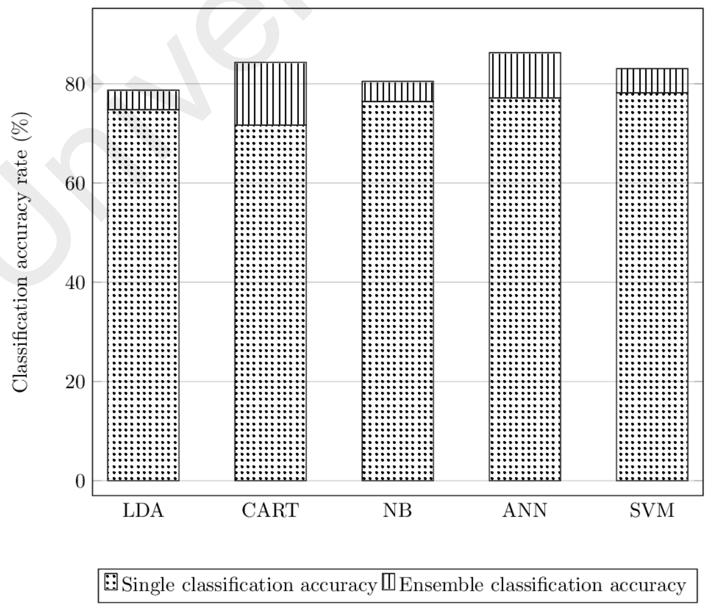


Figure 5.8 Testing error rate (in %) of single classifiers and RLO ensemble methods for sonar data set

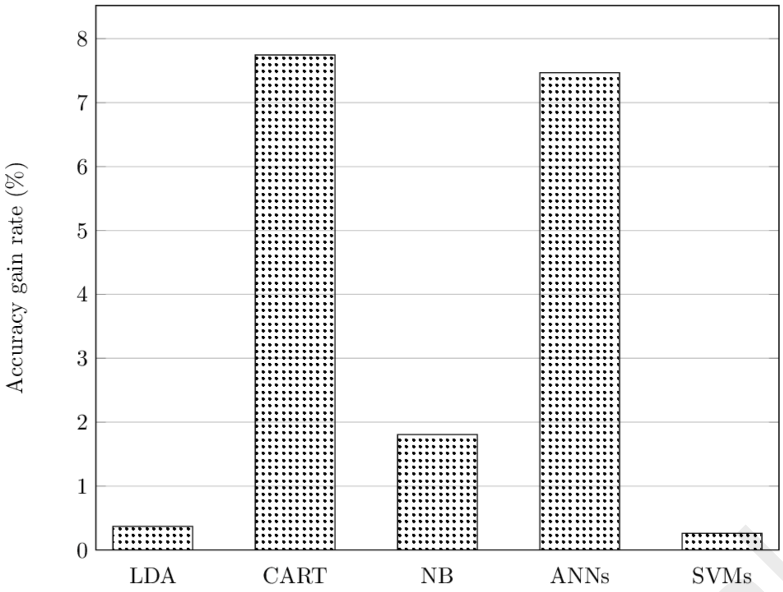


Figure 5.9 Average percentage gains of accuracy (in %) by ensemble methods with respect to single classifiers

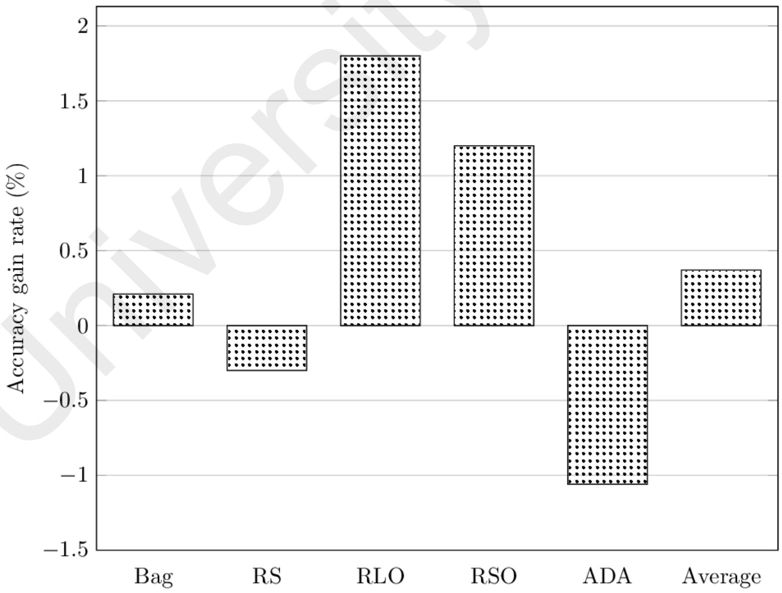


Figure 5.10 Average percentage gains of accuracy (in %) for ensemble LDA with respect to single LDA

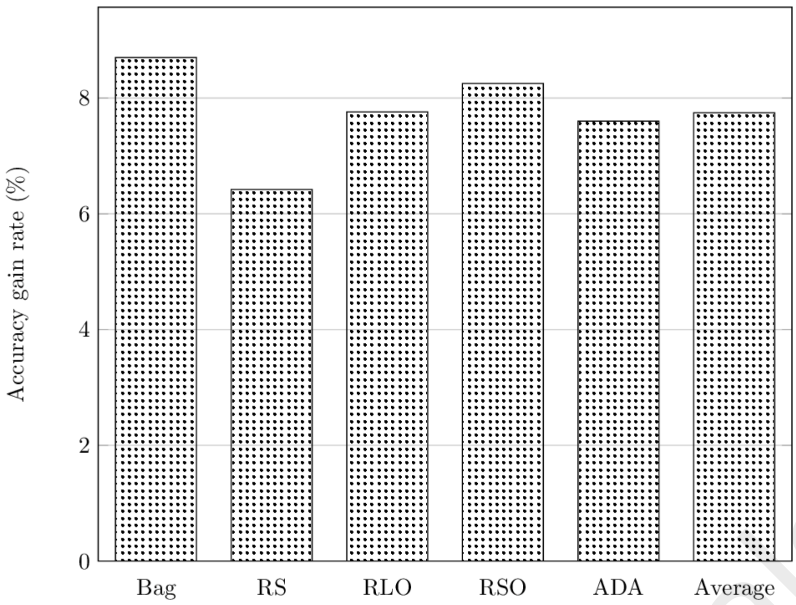


Figure 5.11 Average percentage gains of accuracy (in %) for ensemble CART with respect to single CART

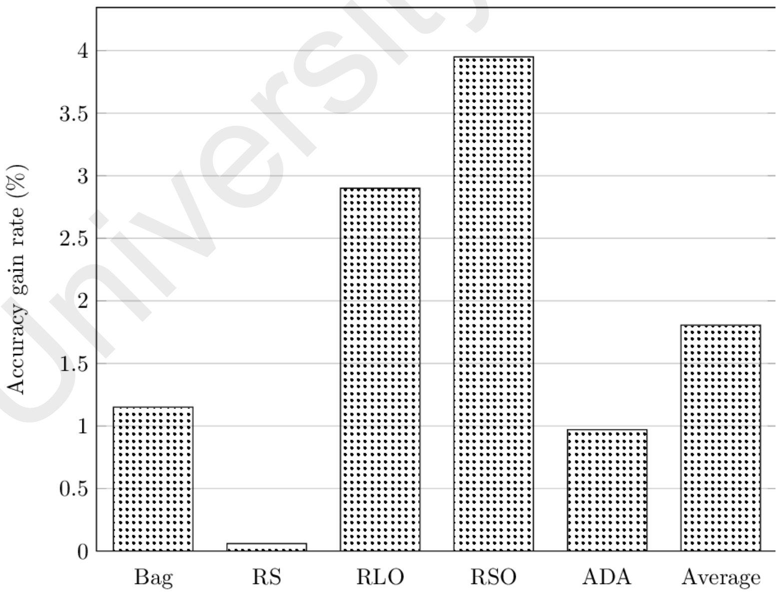


Figure 5.12 Average percentage gains of accuracy (in %) for ensemble NB with respect to single NB

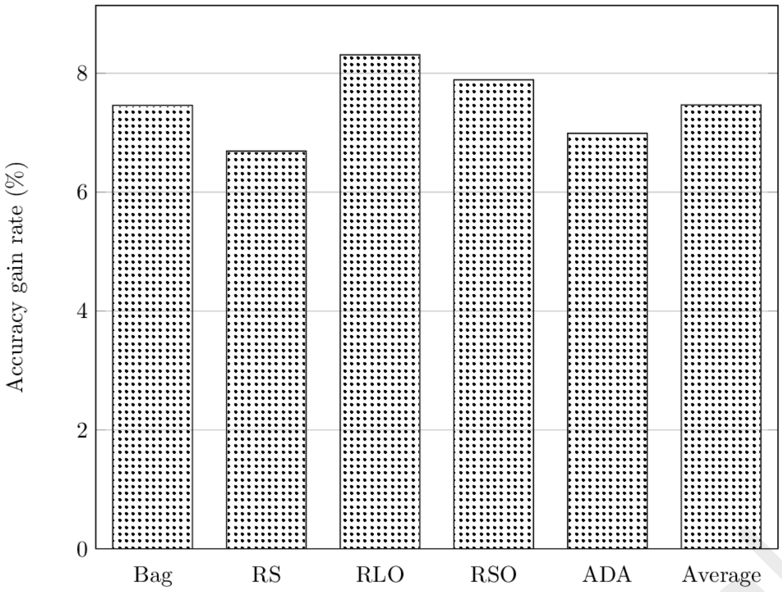


Figure 5.13 Average percentage gains of accuracy (in %) for ensemble ANNs with respect to single ANNs

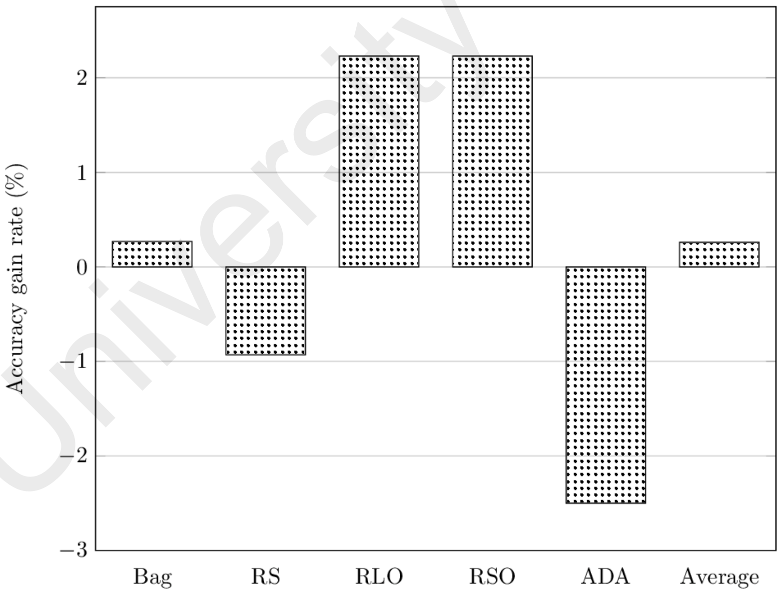


Figure 5.14 Average percentage gains of accuracy (in %) for ensemble SVMs with respect to single SVMs

Table 5.23 Average ranks of testing error rates for the same ensemble method with different base classifiers

| **Algorithm** | **Bag** | **RS** | **RLO** | **RSO** | **ADA** |
|---|---|---|---|---|---|
| LDA | 3.17 | 3.00 | 3.00 | 3.00 | 3.67 |
| CART | 3.33 | 3.50 | 3.83 | 3.67 | 2.33 |
| NB | 3.50 | 3.67 | 3.83 | 3.17 | 3.83 |
| ANN | **2.00** | **1.67** | **2.00** | **2.50** | **1.83** |
| SVM | 3.00 | 3.17 | 2.33 | 2.67 | 3.33 |

Table 5.24 sort of average ranks reported in Table 5.23

| Algorithm | Bag | RS | RLO | RSO | ADA | Average rank |
|---|---|---|---|---|---|---|
| LDA | 3 | 2 | 3 | 3 | 4 | 3 |
| CART | 4 | 4 | 4.5 | 5 | 2 | 3.9 |
| NB | 5 | 5 | 4.5 | 4 | 5 | 4.7 |
| ANN | 1 | 1 | 1 | 1 | 1 | **1** |
| SVM | 2 | 3 | 2 | 2 | 3 | 2.4 |

Table 5.25 Average ranks of testing error rates for single and ensemble classifiers

| **Algorithm** | **Single** | **Ensemble method** |
|---|---|---|
| LDA | 2.33 | 3 |
| CART | 4.33 | 3.9 |
| NB | 3 | 4.7 |
| ANN | 3.33 | **1** |
| SVM | **2** | 2.4 |

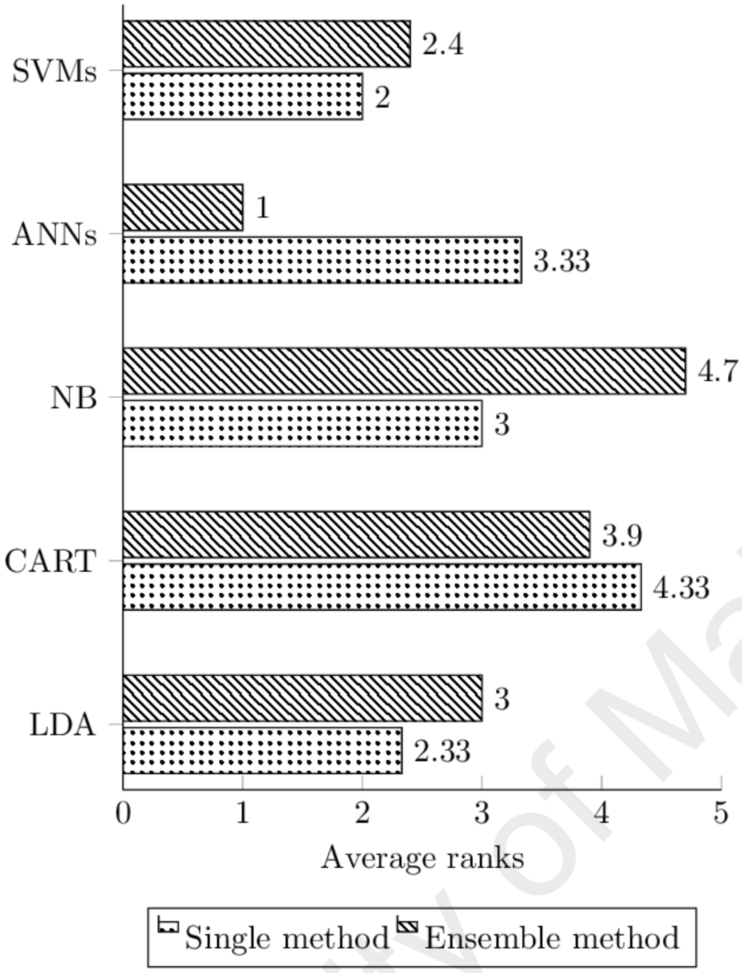


Figure 5.15 Average ranks of testing error rates for single and ensemble classifiers

Table 5.26 Average ranks of testing error rates for different ensemble methods but the same base classifier

| Algorithm | LDA | CART | NB | ANNs | SVMs |
|---|---|---|---|---|---|
| Bag | 2.42 | **1.67** | 3 | 2.83 | 2.83 |
| RS | 3.5 | 4.33 | 3.67 | 3.17 | 3.67 |
| RLO | **2** | 3.17 | **2.5** | **2.5** | **1.83** |
| RSO | 3.17 | 3.33 | **2.5** | 3.17 | 2.67 |
| ADA | 3.92 | 2.5 | 3.33 | 3.33 | 4 |

Table 5.27 sort of average ranks of Table 5.26

| Algorithm | LDA | CART | NB | ANNs | SVMs | Average rank |
|---|---|---|---|---|---|---|
| Bag | 2 | **1** | 3 | 2 | 3 | 2.2 |
| RS | 4 | 5 | 5 | 3.5 | 4 | 4.3 |
| RLO | **1** | 3 | **1.5** | **1** | **1** | **1.5** |
| RSO | 3 | 4 | **1.5** | 3.5 | 2 | 2.8 |
| ADA | 5 | 2 | 4 | 5 | 5 | 4.2 |



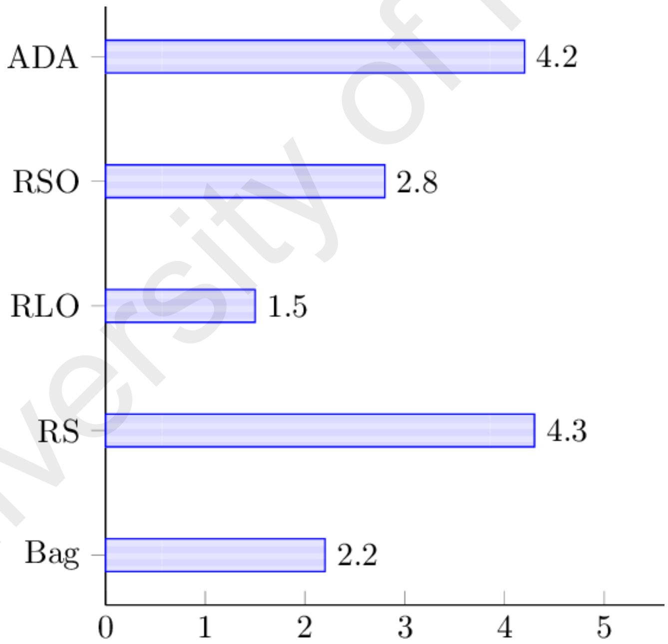


Figure 5.16 Average ranks of testing error rates for ensemble methods (from Table 5.27)

### 5.2.3 Computational Complexity evaluation

The present study aims at designing an accurate ensemble classifier without considering the complexity issue. In this section, however, we provide some results about the computational time of each of the five classifiers considered in this study. The objective of this section is to give an idea about the computational complexity of some algorithms so that it helps to user to choose between them if the complexity measure is really important for the application under consideration.

Tables 5.28 and 5.29 show the computational time needed to build one classifier for each of the five algorithms. In addition, Tables 5.30 and 5.31 represent the computational time spent by each of the classifier to classify one testing pattern. To make the comparison easier, the computational time of each algorithm is divided on the lowest computational time achieved by one of the algorithms for each data set.

We can notice that SVM is the fastest algorithm for the classification task but its computational cost on the training phase (to build the classifier) is not consistent and it is sometimes as in Diabetes and Heart C data sets highly expensive. This may be caused by the complexity's degree of some data sets that might need more time to be approximated by the classifier. On the other hand, the results show that CART is also fast but more stable on both training and classification and this may indicate the reason of CART's widely application in various domains either as single classifier or as base classifier for different ensemble methods. Other classifiers such as LDA and ANN are relatively stable classifiers but less efficient in terms of computational costs compared to CART. Lastly, NB seems more efficient in training than in classification especially in challenging datasets such as Sonar and Heart C.

Actually, it is difficult to make any recommendation based on the computational complexity only as the accuracy of classifiers is generally more important than their computational costs. In fact, it was found that any improvement (even small) in accuracy for some applications like financial prediction systems can be translated into a huge savings (D. West, 2000).

In addition, the computational complexity of the algorithm-as we notice from the results obtained- depends mainly on the data set under study which makes difficult to draw a general conclusion for all classification problem. Furthermore, the recent advances in hardware technology and cloud computing have largely contributed in solving the computational complexity issue (Armbrust et al., 2010).

Table 5.28 Computational time (in seconds) needed to build one classifier for each of the five algorithms

| Dataset | # attributes | # patterns | # classes | Computational time (in seconds) | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | LDA | CART | NB | ANNs | SVMs |
| **Breast W** | 9 | 683 | 2 | 0.628512 | 0.03951 | 0.066506 | 1.575865 | 0.023723 |
| **Diabetes** | 8 | 768 | 2 | 0.014151 | 0.024181 | 0.059617 | 0.101691 | 11.7852 |
| **Glass** | 9 | 214 | 6 | 0.035911 | 0.010342 | 0.136 | 0.117472 | 0.021077 |
| **Heart C** | 13 | 297 | 5 | 0.098415 | 0.015827 | 0.21901 | 0.083033 | 8.356938 |
| **Sonar** | 60 | 208 | 2 | 2.1557 | 0.026235 | 0.373488 | 0.119714 | 0.012418 |
| **Wine** | 13 | 178 | 3 | 0.021053 | 0.008091 | 0.099040 | 0.118307 | 0.553244 |

Table 5.29 Computational time (in units of the lowest computational time for each data set) needed to build one classifier for each of the five algorithms

| Dataset | # attributes | # patterns | # classes | Computational time (in units) | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | LDA | CART | NB | ANNs | SVMs |
| **Breast W** | 9 | 683 | 2 | 26.49 | 1.67 | 2.80 | 66.43 | 1.00 |
| **Diabetes** | 8 | 768 | 2 | 1.00 | 1.71 | 4.21 | 7.19 | 832.84 |
| **Glass** | 9 | 214 | 6 | 3.47 | 1.00 | 13.15 | 11.36 | 2.04 |
| **Heart C** | 13 | 297 | 5 | 6.22 | 1.00 | 13.84 | 5.25 | 528.03 |
| **Sonar** | 60 | 208 | 2 | 173.60 | 2.11 | 30.08 | 9.64 | 1.00 |
| **Wine** | 13 | 178 | 3 | 2.60 | 1.00 | 12.24 | 14.62 | 68.38 |

Table 5.30 Computational time (in seconds) needed to classify one testing pattern for each of the five algorithms

| Dataset | # attributes | # patterns | # classes | Computational time (in seconds) | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | LDA | CART | NB | ANNs | SVMs |
| **Breast W** | 9 | 683 | 2 | 0.001732 | 0.00118 | 0.0115 | 0.007758 | 0.000392 |
| **Diabetes** | 8 | 768 | 2 | 0.002169 | 0.001226 | 0.012623 | 0.007858 | 0.000424 |
| **Glass** | 9 | 214 | 6 | 0.001762 | 0.000925 | 0.022806 | 0.007915 | 0.000389 |
| **Heart C** | 13 | 297 | 5 | 0.002402 | 0.001247 | 0.038763 | 0.007868 | 0.000412 |
| **Sonar** | 60 | 208 | 2 | 0.002772 | 0.00149 | 0.050188 | 0.008191 | 0.000488 |
| **Wine** | 13 | 178 | 3 | 0.001722 | 0.000865 | 0.021672 | 0.007889 | 0.000378 |

Table 5.31 Computational time (in units of the lowest computational time for each data set) needed to classify one testing pattern for each of the five algorithms

| Dataset | # attributes | # patterns | # classes | Computational time (in units) | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | LDA | CART | NB | ANNs | SVMs |
| **Breast W** | 9 | 683 | 2 | 4.42 | 3.01 | 29.31 | 19.77 | 1.00 |
| **Diabetes** | 8 | 768 | 2 | 5.12 | 2.89 | 29.79 | 18.55 | 1.00 |
| **Glass** | 9 | 214 | 6 | 4.53 | 2.38 | 58.60 | 20.34 | 1.00 |
| **Heart C** | 13 | 297 | 5 | 5.83 | 3.03 | 94.08 | 19.10 | 1.00 |
| **Sonar** | 60 | 208 | 2 | 5.69 | 3.05 | 102.94 | 16.80 | 1.00 |
| **Wine** | 13 | 178 | 3 | 4.56 | 2.29 | 57.36 | 20.88 | 1.00 |

### 5.2.4 Comparison between ensemble methods built with different base classifiers and the combination of these ensemble methods

In this section, we compare the proposed design for ensemble methods which combine the outputs of all base classifiers with the commonly used method that used only one base classifier. In addition, we evaluate the usefulness of using GA-based selection method based on two diversity criteria, namely, *double default (DF)* and *difficulty (Diff)*. This selection method aims to choose a subset of the ensemble outputs based on their diversity. As can be seen in Tables 5.37 and 5.38 which summarize Tables 5.32 to 5.36, GA-based selection using Accuracy and double fault measures (S-(Acc+DF)) has successfully achieved the best results in three ensemble methods, namely, Bagging, RS and RLO. In RSO and AdaBoost, the two methods that have the best accuracy are GA-based selection using accuracy only (S-Acc) and the method which considers all the outputs without selection (All-MV), respectively. As a result, the idea of combining the outputs of ensemble methods with different base classifiers seems better than using one ensemble method with a given base classifier. One possible interpretation of this result might be that combining different base classifiers may create more diversity in the ensemble methods. In addition, using GA-based selection method with appropriate measure (such as accuracy and double default) is better than considers the outputs of all the ensemble methods.  On the other hand, GA-based selection using the accuracy criterion only (S-Acc) is less accurate than considering all the output (All-MV) except in RLO and RSO. This may due, as we mentioned before, that RLO and RSO are promoting diversity in their ensemble members and hence they don't need the diversity created by GA-selection. This can be especially visible in RSO where unlike other ensemble methods, none of the selection methods based on the diversity criteria got the first or even the second best testing error rate.

Table 5.32 Average testing error rates (in %) of Bagging using different methods[20]

| | LDA | CART | NB | ANN | SVM | All | S-Acc | S-(Acc+DF) | S-(Acc+Diff) |
|---|---|---|---|---|---|---|---|---|---|
| Breast W | 4 | 3.51 | 3.18 | 2.99 | 3.37 | 3.07 | 3.12 | 2.96 | 2.99 |
| Diabetes | 22.59 | 23.87 | 25.65 | 22.9 | 22.54 | 22.77 | 23.02 | 22.8 | 22.61 |
| Glass | 37.99 | 25.12 | 36.48 | 36.38 | 38.21 | 32.57 | 30.41 | 27.86 | 29.2 |
| Heart C | 40.95 | 44.46 | 44.28 | 42.62 | 39.95 | 40.89 | 41.1 | 41.38 | 41.5 |
| Sonar | 24.98 | 19.43 | 24.02 | 16.44 | 22.32 | 19.23 | 18.22 | 17.54 | 17.92 |
| Wine | 1.51 | 3.67 | 2.55 | 1.91 | 4.57 | 1.69 | 1.74 | 1.46 | 1.35 |

Table 5.33 Average testing error rates (in %) of Random Subspace (RS) using different methods

| | LDA | CART | NB | ANN | SVM | All | S-Acc | S-(Acc+DF) | S-(Acc+Diff) |
|---|---|---|---|---|---|---|---|---|---|
| Breast W | 4.11 | 3.67 | 3.51 | 2.97 | 3.31 | 3.07 | 3.12 | 2.97 | 3.04 |
| Diabetes | 23.2 | 26.71 | 27.04 | 23.15 | 23.46 | 23.25 | 23.29 | 23.11 | 23.17 |
| Glass | 39.32 | 25.75 | 40.18 | 37.48 | 40.6 | 34.51 | 31.8 | 29.18 | 30.05 |
| Heart C | 41.32 | 45.86 | 43.35 | 43.66 | 41.73 | 43.04 | 43.27 | 43.37 | 43.2 |
| Sonar | 24.41 | 20.93 | 24.14 | 17.03 | 22.39 | 19.33 | 19.18 | 18.85 | 18.85 |
| Wine | 1.68 | 6.93 | 2.14 | 1.46 | 3.74 | 2.02 | 2.09 | 1.92 | 2.09 |

Table 5.34 Average testing error rates (in %) of Random Linear Oracle (RLO) using different methods

| | LDA | CART | NB | ANN | SVM | All | S-Acc | S-(Acc+DF) | S-(Acc+Diff) |
|---|---|---|---|---|---|---|---|---|---|
| Breast W | 3.37 | 3.53 | 3.26 | 2.97 | 3.24 | 2.99 | 2.93 | 2.93 | 2.94 |
| Diabetes | 22.97 | 24.85 | 24.72 | 22.93 | 22.94 | 22.87 | 22.71 | 22.63 | 22.61 |
| Glass | 34.29 | 26.71 | 35.02 | 34.97 | 34.15 | 27.63 | 26.48 | 24.57 | 25.06 |
| Heart C | 41.75 | 46.9 | 43.61 | 43.14 | 41.36 | 41.65 | 42.15 | 42.18 | 42.62 |
| Sonar | 21.29 | 15.68 | 19.52 | 13.74 | 16.96 | 14.19 | 14.04 | 13.7 | 14.03 |
| Wine | 1.7 | 5.58 | 2.78 | 2.24 | 3.59 | 1.63 | 1.58 | 1.46 | 1.35 |

[20] To get the full names of the methods listed in these tables, please refer to Appendix B, Table B.1

Table 5.35 Average testing error rates (in %) of Random Sphere Oracle (RSO) using different methods

| | LDA | CART | NB | ANN | SVM | All | S-Acc | S-(Acc+DF) | S-(Acc+Diff) |
|---|---|---|---|---|---|---|---|---|---|
| Breast W | 3.63 | 3.57 | 3.18 | 2.99 | 3.31 | 2.9 | 2.82 | 2.82 | 2.8 |
| Diabetes | 23.37 | 25.27 | 24.74 | 23.66 | 23.16 | 23.33 | 23.42 | 23.41 | 23.45 |
| Glass | 34.71 | 26.96 | 32.37 | 36.11 | 32.71 | 28.62 | 27.23 | 26.55 | 26.17 |
| Heart C | 41.97 | 45.68 | 43.74 | 42.64 | 42.5 | 42.01 | 42 | 42.82 | 42.72 |
| Sonar | 22.44 | 14.07 | 17.41 | 13.95 | 16.91 | 14.47 | 13.99 | 14.42 | 14.56 |
| Wine | 1.92 | 6.31 | 3.5 | 2.07 | 3.69 | 2.25 | 2.25 | 2.31 | 2.31 |

Table 5.36 Average testing error rates (in %) of Adaboost using different methods[21]

| | LDA | CART | NB | ANN | SVM | All |
|---|---|---|---|---|---|---|
| Breast W | 4.28 | 3 | 3.39 | 3.09 | 3.18 | 2.96 |
| Diabetes | 26.44 | 24.2 | 26.21 | 25.21 | 24.79 | 23.63 |
| Glass | 38.76 | 30.3 | 36.8 | 35.75 | 43.24 | 33.07 |
| Heart C | 42.27 | 42.94 | 42.81 | 41.14 | 41.64 | 41.98 |
| Sonar | 24.15 | 20.5 | 25.84 | 19.29 | 24.05 | 19.76 |
| Wine | 1.51 | 4.35 | 2.37 | 1.69 | 5.24 | 1.86 |

Table 5.37 Average ranks of testing error rates for different methods on the same ensemble method

| **Algorithm** | **Bag** | **RS** | **RLO** | **RSO** | **ADA** |
|---|---|---|---|---|---|
| LDA | 5.67 | 5.33 | 6.50 | 5.17 | 4.50 |
| CART | 6.67 | 6.83 | 7.67 | 6.83 | 3.17 |
| NB | 7.50 | 7.50 | 7.83 | 7.17 | 4.83 |
| ANN | 4.83 | 3.33 | 5.33 | 4.83 | 2.33 |
| SVM | 5.67 | 6.50 | 5.67 | 5.67 | 4.17 |
| All-MV | 4.00 | 4.33 | 4.17 | 3.75 | **2.00** |
| S-Acc | 4.83 | 4.83 | 3.17 | **3.25** | |
| S-(Acc+DF) | **2.83** | **2.67** | **2.00** | 4.17 | |
| S-(Acc+Diff) | 3.00 | 3.67 | 2.67 | 4.17 | |

[21] Unlike the other ensemble methods, Adaboost produces weighted outputs which makes applying the other methods infeasible.

Table 5.38 sort of average ranks of Table 5.33

| Algorithm | Bag | RS | RLO | RSO | ADA | Average rank |
|---|---|---|---|---|---|---|
| LDA | 6 | 6 | 7 | 6 | 5 | 6 |
| CART | 8 | 8 | 8 | 8 | 3 | 7 |
| NB | 9 | 9 | 9 | 9 | 6 | 8.4 |
| ANN | 4 | 2 | 5 | 5 | 2 | 3.6 |
| SVM | 6 | 7 | 6 | 7 | 4 | 6 |
| All-MV | 3 | 4 | 4 | 2 | **1** | 2.8 |
| S-Acc | 4 | 5 | 3 | **1** | | 3.25 |
| S-(Acc+DF) | **1** | **1** | **1** | 3 | | 1.5 |
| S-(Acc+Diff) | 2 | 3 | 2 | 3 | | 2.5 |

### 5.2.5 Comparison between all methods

This is the last step of comparison which aims to select the most accurate ensemble method which will be used with the fuzzy rule-based system. In addition to the previously discussed ensemble methods and single classifiers, we added another method called Random Forest (RF) (Leo Breiman, 2001), a well-known ensemble method based on CART.

By considering all the 48 methods, we can see from Tables 5.40 and 5.41 which ranks the methods according to their results listed in Table 5.39 that the top 10 methods (which are shown in Figure 5.17) are those which combine all the base classifiers for designing the ensemble methods. In addition, the three RLO methods that used GA-based selection approaches rank as the three best methods.

Specifically, the use of GA-based selection method with accuracy and double fault as measures for the fitness function ranks as the best method. In this case, RLO(S-(Acc+DF)) was chosen as the ensemble method to be employed for the fuzzy-ensemble method.

Table 5.39 Average testing error rates (in %) for 48 methods[22]

| | Breast W | Diabetes | Glass | Heart C | Sonar | Wine |
|---|---|---|---|---|---|---|
| RF | 2.93 | 23.9 | 20.7 | 42.81 | 16.72 | 1.86 |
| S(LDA) | 4.01 | 22.63 | 38.36 | 41.12 | 25.26 | 1.46 |
| S(CART) | 5.08 | 29.31 | 31.39 | 51.44 | 28.36 | 9.86 |
| S(NB) | 3.25 | 26.48 | 40.51 | 44.15 | 23.59 | 2.26 |
| S(ANN) | 3.53 | 25.84 | 50.47 | 42.07 | 22.86 | 6.07 |
| S(SVM) | 3.43 | 22.71 | 38.53 | 40.91 | 21.84 | 4.41 |
| Bag(LDA) | 4 | 22.59 | 37.99 | 40.95 | 24.98 | 1.51 |
| Bag(CART) | 3.51 | 23.87 | 25.12 | 44.46 | 19.43 | 3.67 |
| Bag(NB) | 3.18 | 25.65 | 36.48 | 44.28 | 24.02 | 2.55 |
| Bag(ANN) | 2.99 | 22.9 | 36.38 | 42.62 | 16.44 | 1.91 |
| Bag(SVM) | 3.37 | 22.54 | 38.21 | 39.95 | 22.32 | 4.57 |
| Bag(All) | 3.07 | 22.77 | 32.57 | 40.89 | 19.23 | 1.69 |
| Bag(S-Acc) | 3.12 | 23.02 | 30.41 | 41.1 | 18.22 | 1.74 |
| Bag(S-(Acc+DF)) | 2.96 | 22.8 | 27.86 | 41.38 | 17.54 | 1.46 |
| Bag(S-(Acc+Diff)) | 2.99 | 22.61 | 29.2 | 41.5 | 17.92 | 1.35 |
| RS(LDA) | 4.11 | 23.2 | 39.32 | 41.32 | 24.41 | 1.68 |
| RS(CART) | 3.67 | 26.71 | 25.75 | 45.86 | 20.93 | 6.93 |
| RS(NB) | 3.51 | 27.04 | 40.18 | 43.35 | 24.14 | 2.14 |
| RS(ANN) | 2.97 | 23.15 | 37.48 | 43.66 | 17.03 | 1.46 |
| RS(SVM) | 3.31 | 23.46 | 40.6 | 41.73 | 22.39 | 3.74 |
| RS(All) | 3.07 | 23.25 | 34.51 | 43.04 | 19.33 | 2.02 |
| RS(S-Acc) | 3.12 | 23.29 | 31.8 | 43.27 | 19.18 | 2.09 |
| RS(S-(Acc+DF)) | 2.97 | 23.11 | 29.18 | 43.37 | 18.85 | 1.92 |
| RS(S-(Acc+Diff)) | 3.04 | 23.17 | 30.05 | 43.2 | 18.85 | 2.09 |
| RLO(LDA) | 3.37 | 22.97 | 34.29 | 41.75 | 21.29 | 1.7 |
| RLO(CART) | 3.53 | 24.85 | 26.71 | 46.9 | 15.68 | 5.58 |
| RLO(NB) | 3.26 | 24.72 | 35.02 | 43.61 | 19.52 | 2.78 |
| RLO(ANN) | 2.97 | 22.93 | 34.97 | 43.14 | 13.74 | 2.24 |
| RLO(SVM) | 3.24 | 22.94 | 34.15 | 41.36 | 16.96 | 3.59 |
| RLO(All) | 2.99 | 22.87 | 27.63 | 41.65 | 14.19 | 1.63 |
| RLO(S-Acc) | 2.93 | 22.71 | 26.48 | 42.15 | 14.04 | 1.58 |
| **RLO(S-(Acc+DF))** | 2.93 | 22.63 | 24.57 | 42.18 | 13.7 | 1.46 |
| RLO(S-(Acc+Diff)) | 2.94 | 22.61 | 25.06 | 42.62 | 14.03 | 1.35 |
| RSO(LDA) | 3.63 | 23.37 | 34.71 | 41.97 | 22.44 | 1.92 |
| RSO(CART) | 3.57 | 25.27 | 26.96 | 45.68 | 14.07 | 6.31 |
| RSO(NB) | 3.18 | 24.74 | 32.37 | 43.74 | 17.41 | 3.5 |
| RSO(ANN) | 2.99 | 23.66 | 36.11 | 42.64 | 13.95 | 2.07 |
| RSO(SVM) | 3.31 | 23.16 | 32.71 | 42.5 | 16.91 | 3.69 |
| RSO(All) | 2.9 | 23.33 | 28.62 | 42.01 | 14.47 | 2.25 |
| RSO(S-Acc) | 2.82 | 23.42 | 27.23 | 42 | 13.99 | 2.25 |
| RSO(S-(Acc+DF)) | 2.82 | 23.41 | 26.55 | 42.82 | 14.42 | 2.31 |
| RSO(S-(Acc+Diff)) | 2.8 | 23.45 | 26.17 | 42.72 | 14.56 | 2.31 |
| ADA(LDA) | 4.28 | 26.44 | 38.76 | 42.27 | 24.15 | 1.51 |
| ADA(CART) | 3 | 24.2 | 30.3 | 42.94 | 20.5 | 4.35 |
| ADA(NB) | 3.39 | 26.21 | 36.8 | 42.81 | 25.84 | 2.37 |
| ADA(ANN) | 3.09 | 25.21 | 35.75 | 41.14 | 19.29 | 1.69 |
| ADA(SVM) | 3.18 | 24.79 | 43.24 | 41.64 | 24.05 | 5.24 |
| ADA(All) | 2.96 | 23.63 | 33.07 | 41.98 | 19.76 | 1.86 |

[22] To get the full name of these methods, see Appendix B, Table B.1

Table 5.40 Average ranks of testing error rates for 48 methods

| Method | Average rank |
|---|---|
| RF | 16.67 |
| S(LDA) | 24.50 |
| S(CART) | 43.33 |
| S(NB) | 38.17 |
| S(ANN) | 38.75 |
| S(SVM) | 27.08 |
| Bag(LDA) | 23.42 |
| Bag(CART) | 30.42 |
| Bag(NB) | 36.33 |
| Bag(ANN) | 19.67 |
| Bag(SVM) | 25.25 |
| Bag(All) | 15.33 |
| Bag(S-Acc) | 16.58 |
| Bag(S-(Acc+DF)) | 11.00 |
| Bag(S-(Acc+Diff)) | 11.25 |
| RS(LDA) | 28.83 |
| RS(CART) | 36.50 |
| RS(NB) | 38.75 |
| RS(ANN) | 21.42 |
| RS(SVM) | 32.58 |
| RS(All) | 25.25 |
| RS(S-Acc) | 25.17 |
| RS(S-(Acc+DF)) | 20.67 |
| RS(S-(Acc+Diff)) | 22.83 |
| RLO(LDA) | 22.92 |
| RLO(CART) | 31.58 |
| RLO(NB) | 33.00 |
| RLO(ANN) | 19.50 |
| RLO(SVM) | 21.50 |
| RLO(All) | 11.58 |
| RLO(S-Acc) | 9.42 |
| **RLO(S-(Acc+DF))** | 6.83 |
| RLO(S-(Acc+Diff)) | 7.75 |
| RSO(LDA) | 28.08 |
| RSO(CART) | 31.50 |
| RSO(NB) | 29.67 |
| RSO(ANN) | 21.92 |
| RSO(SVM) | 25.25 |
| RSO(All) | 16.42 |
| RSO(S-Acc) | 15.00 |
| RSO(S-(Acc+DF)) | 17.83 |
| RSO(S-(Acc+Diff)) | 17.42 |
| ADA(LDA) | 34.42 |
| ADA(CART) | 28.83 |
| ADA(NB) | 37.08 |
| ADA(ANN) | 23.08 |
| ADA(SVM) | 34.33 |
| ADA(All) | 21.33 |

Table 5.41 Sort of average ranks of 48 methods reported in Table 5.36

| Algorithm | Ranks |
|---|---|
| **RLO(S-(Acc+DF))** | 1 |
| RLO(S-(Acc+Diff )) | 2 |
| RLO(S-Acc) | 3 |
| Bag(S-(Acc+DF)) | 4 |
| Bag(S-(Acc+Diff )) | 5 |
| RLO(All) | 6 |
| RSO(S-Acc) | 7 |
| Bag(All) | 8 |
| RSO(All) | 9 |
| Bag(S-Acc) | 10 |
| RF | 11 |
| RSO(S-(Acc+Diff )) | 12 |
| RSO(S-(Acc+DF)) | 13 |
| RLO(ANN) | 14 |
| Bag(ANN) | 15 |
| RS(S-(Acc+DF)) | 16 |
| ADA(All) | 17 |
| RS(ANN) | 18 |
| RLO(SVM) | 19 |
| RSO(ANN) | 20 |
| RS(S-(Acc+Diff)) | 21 |
| RLO(LDA) | 22 |
| ADA(ANN) | 23 |
| Bag(LDA) | 24 |
| S(LDA) | 25 |
| RS(S-Acc) | 26 |
| Bag(SVM) | 28 |
| RS(All) | 28 |
| RSO(SVM) | 28 |
| S(SVM) | 30 |
| RSO(LDA) | 31 |
| RS(LDA) | 32.5 |
| ADA(CART) | 32.5 |
| RSO(NB) | 34 |
| Bag(CART) | 35 |
| RSO(CART) | 36 |
| RLO(CART) | 37 |
| RS(SVM) | 38 |
| RLO(NB) | 39 |
| ADA(SVM) | 40 |
| ADA(LDA) | 41 |
| Bag(NB) | 42 |
| RS(CART) | 43 |
| ADA(NB) | 44 |
| S(NB) | 45 |
| S(ANN) | 46.5 |
| RS(NB) | 46.5 |
| S(CART) | 48 |

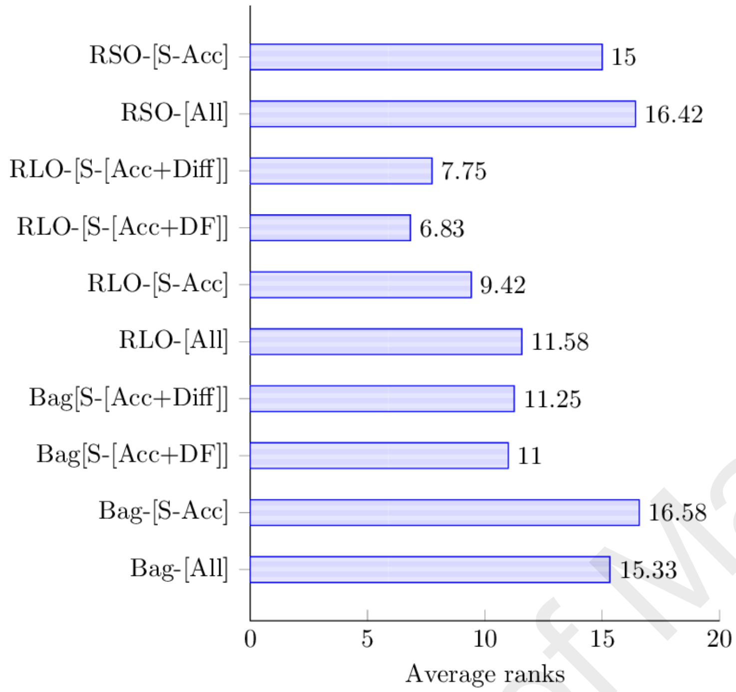


Figure 5.17 Average ranks of testing error rates for the top 10 ensemble methods

### 5.2.6 Comparison with a benchmark fuzzy-based ensemble method

For the purpose of comparison, we select a related work proposed in (Trawinski, Cordon, Sanchez, et al., 2013) which uses RLO as an ensemble method and FURIA, a fuzzy rule-based method, as a base classifier. From Table 5.42, which shows the classification accuracy for both our selected ensemble method RLO(S-(Acc+DF)) and the fuzzy-based ensemble method, we can see that our ensemble method is more accurate than the fuzzy-based ensemble method which confirms the previous observation that the selection of base classifier is an important factor for the accuracy of the ensemble method. So, the accuracy of our ensemble method is competitive with existing methods.

Table 5.42 Average testing error rates (in %) for RLO(S-(Acc+DF)) and the method reported in (Trawinski et al., 2013)

| | Reported in (Trawinski, Cordon, Sanchez, et al., 2013) | RLO(S-(Acc+DF )) |
|---|---|---|
| Breast W | 4.09 | 2.93 |
| Diabetes | 23.98 | 22.63 |
| Glass | 28.32 | 24.57 |
| Heart C | / | / |
| Sonar | 20.77 | 13.7 |
| Wine | 3.03 | 1.46 |
| Average rank | **2** | **1** |

### 5.3 Fuzzy-Ensemble method

In the previous phases, we have evaluated different methods for both fuzzy rule-based systems and ensemble methods and based on the results obtained, we selected Proposal2 for fuzzy rule-based system and RLO(S-(Acc+DF)) for the ensemble method.

In this phase, we study the performance of the combined fuzzy-ensemble method and specifically, we focus on two functionalities, namely, classification and interpretation.

#### 5.3.1 Comparison between Method1 and Method2

As we explained in chapter4, the fuzzy rule-based system is used for both prediction and interpretation. But, in some cases where the pattern classification certainties are lower than a threshold value, the fuzzy rule-based system classification is rejected and the ensemble method is called to perform the classification. Two methods are used to calculate the threshold values $\theta_1$ and $\theta_2$. Tables 5.43 and 5.44 show the results of testing error rates produced using the thresholds $\theta_1$ and $\theta_2$, respectively. As we can see from the two tables, for both $\theta_1$ and $\theta_2$, testing error rates of the fuzzy rule-based systems with the support of ensemble methods (i.e. fuzzy-ensemble method) were improved for all the data sets compared with testing error rates of the fuzzy systems without support. We can notice also that in uncovered testing patterns, which are considered as incorrectly predicted by the fuzzy system (because the predicted output, in this case, is empty or non-identifiable class), the accuracy increased from 0 in the case of fuzzy system to 50%-100% with support of the ensemble method. In addition, by examining the results from the two tables, we can see that the performance of the ensemble method is better than the fuzzy system on the patterns that have certainty grade less than the threshold value. This result gives justification of using the ensemble method for these patterns. Furthermore, the performance of the fuzzy system on the patterns whose certainty grade is greater than the threshold value is better than those with certainty values less than the threshold for both $\theta_1$ and $\theta_2$. This confirms what we assumed before that the performance of fuzzy system decreases when the confidence value goes towards zero.

Table 5.43 Average testing error rates (in %) for fuzzy-ensemble method using $\theta_1$

| | Threshold $\theta_1$ | Testing error rate of fuzzy system on patterns with $\mu_{R_w^j}(x_j).r_j^w > \theta_1$ | Testing error rate on uncovered patterns | | Testing error rate on patterns with $\mu_{R_w^j}(x_j).r_j^w < \theta_1$ | | Testing error rate of fuzzy system without ensemble support (all patterns) | Testing error rate of fuzzy system with ensemble support (all patterns) |
|---|---|---|---|---|---|---|---|---|
| | | | fuzzy | ensemble | fuzzy | ensemble | | |
| Breast W | 0.3606 | 1.26 | 100 | 2.38 | 14.28 | 11.54 | 3.82 | 3.16 |
| Diabetes | 0.102 | 16.35 | 100 | 27.27 | 33.81 | 31.26 | 24.56 | 23.28 |
| Glass | 0.0342 | 32.16 | 100 | 49.21 | 40.06 | 26.57 | 38.1 | 28.71 |
| Heart C | 0.0285 | 68.58 | 100 | 12.5 | 40.13 | 37.9 | 46.025 | 43.7 |
| Sonar | 0.0797 | 16.98 | 100 | 0 | 26.8 | 16.23 | 22.85 | 16.28 |
| Wine | 0.1923 | 1.84 | 100 | 0 | 8.15 | 2.23 | 5.03 | 2.31 |

Table 5.44 Average testing error rates (in %) for fuzzy-ensemble method using $\theta_2$

| | Threshold $\theta_2$ | Testing error rate on patterns with $\mu_{R_w^j}(x_j).r_j^w - \mu_{R_{w'}^j}(x_j).r_j^{w'} > \theta_2$ | Testing error rate on uncovered patterns | | Testing error rate on patterns with $\mu_{R_w^j}(x_j).r_j^w - \mu_{R_{w'}^j}(x_j).r_j^{w'} < \theta_2$ | | Testing error rate of fuzzy system without ensemble support (all patterns) | Testing error rate of fuzzy system with ensemble support (all patterns) |
|---|---|---|---|---|---|---|---|---|
| | | fuzzy | fuzzy | ensemble | fuzzy | ensemble | | |
| Breast W | 0.2482 | 1.88 | 100 | 2.38 | 6.72 | 5.46 | 3.82 | 3.08 |
| Diabetes | 0.0547 | 16.9 | 100 | 27.27 | 32.89 | 29.67 | 24.56 | 22.87 |
| Glass | 0.0264 | 34.07 | 100 | 49.21 | 39.51 | 26.15 | 38.1 | 28.62 |
| Heart C | 0.0239 | 67.26 | 100 | 12.5 | 43.55 | 40.17 | 46.025 | 43.74 |
| Sonar | 0.0475 | 16.2 | 100 | 0 | 26.69 | 15.85 | 22.85 | 16.15 |
| Wine | 0.0682 | 1.31 | 100 | 0 | 16.61 | 4 | 5.03 | 1.92 |

To select which of two methods, Method1 and Method2, is better to use for threshold calculation, we make a comparison between them using two criteria: testing error rate and local interpretability rate expressed as the number of correctly classified testing patterns that have local interpretability over the total number of testing patterns. As we explained in Chapter4, we mean by patterns with local interpretability those which are covered, or simply the compatibility value of their winner rule is greater than zero. These patterns have local interpretability given by the winner rule.

Table 5.45 shows that Method2, used to calculate threshold $\theta_2$, achieved the best results in both testing error rate and local interpretability rate. As a result, we recommend the use of Method2 for threshold calculation and its corresponding rejection method for calling the ensemble method.

Table 5.45 Average testing error rate (in %) and local interpretability rate (in %) for Method1 and Method2

| | Avg. testing error rate (in %) | | Local interpretability rate (in %) | |
|---|---|---|---|---|
| | Method2 | Method1 | Method2 | Method1 |
| Breast W | 3.08 | 3.16 | 96.58 | 96.49 |
| Diabetes | 22.87 | 23.28 | 77.02 | 76.61 |
| Glass | 28.62 | 28.71 | 70.50 | 70.41 |
| Heart C | 43.74 | 43.7 | 55.35 | 55.39 |
| Sonar | 16.15 | 16.28 | 83.76 | 83.63 |
| Wine | 1.92 | 2.31 | 97.91 | 97.52 |
| Average rank | **1.17** | 1.83 | **1.17** | 1.83 |

### 5.3.2 Comparison between fuzzy-ensemble method, fuzzy rule-based system and ensemble method

After selecting Method2 for the fuzzy-ensemble method, it is desirable to make comparisons between the constituent methods, namely, fuzzy rule-based system and ensemble method with the combined fuzzy-ensemble method.

For fuzzy rule-based system and fuzzy-ensemble method, we can notice from Table 5.46 that the average local interpretability among the correctly classified patterns is 100% for the fuzzy rule-based system and 99.40% for the fuzzy-ensemble method. That is 100% and 99.40% of the correctly classified testing patterns by the fuzzy rule-based system and the fuzzy-ensemble method, respectively, are covered and thus interpretable patterns. But if we take the real measure used in comparison which is the rate of local interpretability among all the testing patterns, we find that this rate is lower or at most the same as the rate of testing accuracy. For example, in Glass data, the testing accuracy rate for fuzzy rule-based system and fuzzy-ensemble method is 61.9% and 71.38%, respectively. Local interpretability rate in the same data set is 61.9% for fuzzy rule-based system which represents 100% of the correctly classified testing patterns while the local interpretability rate for fuzzy-ensemble method is 70.50% which is 98.77% of the correctly classified testing patterns. This means that the number of testing patterns that are both correctly classified by the fuzzy-ensemble method and have local interpretations is higher than in the case of the fuzzy rule-based system. Thus, the fuzzy-ensemble method is not only better than the fuzzy rule-based system in testing error rates as can be seen from Figure 5.18 but also in local interpretability rates. We should remind that for global interpretation, both of the methods are equal because they use fuzzy rule-based system to understand and interpret the global behaviour of the system. This suggests that using fuzzy-ensemble method is more useful for the user as he/she can get more accuracy and even more correct local interpretations.

In the case of ensemble and fuzzy-ensemble methods comparison, Table 5.47 shows the classical trade-off between the accuracy and interpretability. That is, while ensemble method is more accurate than fuzzy-ensemble method (see Figure 5.18), the latter is more interpretable compared to the black-box ensemble method. But an interesting improvement to the trade-off was made by fuzzy-ensemble method which, as shown in Table 5.48 and Figure 5.19, has successfully retained an average of 76.77% of the accuracy gains obtained by ensemble method relative to the fuzzy rule-based system.

The rate of retained accuracy can be calculated by:

$$Rate\ of\ retained\ accuracy = 100 \times \frac{Acc_{fuzzy-ensemble} - Acc_{fuzzy}}{Acc_{ensemble} - Acc_{fuzzy}} \tag{5.1}$$

Where $Acc_{ensemble}$ , $Acc_{fuzzy-ensemble}$, $Acc_{fuzzy}$ are the classification rate of the ensemble method, fuzzy-ensemble method and fuzzy rule-based system, respectively. This suggests that while fuzzy-ensemble method maintains to a great extend the superiority of the ensemble method accuracy over the fuzzy rule-based system, it did not lose its interpretability compared to the fuzzy rule-based system; but it is in fact improved its local interpretability rate.

Table 5.46 Average testing error rate (in %) and local interpretability rate (in %) for fuzzy system and fuzzy-ensemble method

| | Avg. testing error rate (in %) | | Avg. testing accuracy rate (in %) | | Local interpretability rate (in %) among all the testing patterns | | Local interpretability rate among the correctly classified patterns (in %) | |
|---|---|---|---|---|---|---|---|---|
| | Fuzzy system | fuzzy-ensemble method | Fuzzy system | fuzzy-ensemble method | Fuzzy system | Fuzzy-ensemble method | Fuzzy system | Fuzzy-ensemble method |
| Breast W | 3.82 | 3.08 | 96.18 | 96.92 | 96.18 | 96.58 | 100 | 99.65 |
| Diabetes | 24.56 | 22.87 | 75.44 | 77.13 | 75.44 | 77.02 | 100 | 99.86 |
| Glass | 38.1 | 28.62 | 61.9 | 71.38 | 61.9 | 70.50 | 100 | 98.77 |
| Heart C | 46.03 | 43.74 | 53.97 | 56.26 | 53.97 | 55.35 | 100 | 98.38 |
| Sonar | 22.85 | 16.15 | 77.15 | 83.85 | 77.15 | 83.76 | 100 | 99.89 |
| Wine | 5.03 | 1.92 | 94.97 | 98.08 | 94.97 | 97.91 | 100 | 99.83 |
| Average | 2 | 1 | 2 | 1 | 2 | 1 | **100** | **99.40** |

Table 5.47 Average testing error rate (in %) and local interpretability rate (in %) for ensemble method and fuzzy-ensemble method

| | Avg. testing error rate (in %) | | Local interpretability rate (in %) | |
|---|---|---|---|---|
| | ensemble method | fuzzy-ensemble method | ensemble method | Fuzzy-ensemble method |
| Breast W | 2.93 | 3.08 | 0 | 96.58 |
| Diabetes | 22.63 | 22.87 | 0 | 77.02 |
| Glass | 24.57 | 28.62 | 0 | 70.50 |
| Heart C | 42.18 | 43.74 | 0 | 55.35 |
| Sonar | 13.7 | 16.15 | 0 | 83.76 |
| Wine | 1.46 | 1.92 | 0 | 97.91 |
| Average ranks | 1 | 2 | 2 | 1 |

Table 5.48 Testing error rate (in %) for the fuzzy rule-based system, ensemble method and fuzzy-ensemble and the rate of retained accuracy by the fuzzy-ensemble method that was gained by the ensemble method relative to the fuzzy rule-based system.

| | Avg. testing error rate (in %) | | | Rate of retained accuracy by fuzzy-ensemble method |
|---|---|---|---|---|
| | Fuzzy system | Ensemble method | Fuzzy-Ensemble method | |
| Breast W | 3.82 | 2.93 | 3.08 | 83.15 |
| Diabetes | 24.56 | 22.63 | 22.87 | 87.56 |
| Glass | 38.1 | 24.57 | 28.62 | 70.07 |
| Heart C | 46.03 | 42.18 | 43.74 | 59.48 |
| Sonar | 22.85 | 13.7 | 16.15 | 73.22 |
| Wine | 5.03 | 1.46 | 1.92 | 87.11 |
| Average rate of retained accuracy | | | | 76.77 |

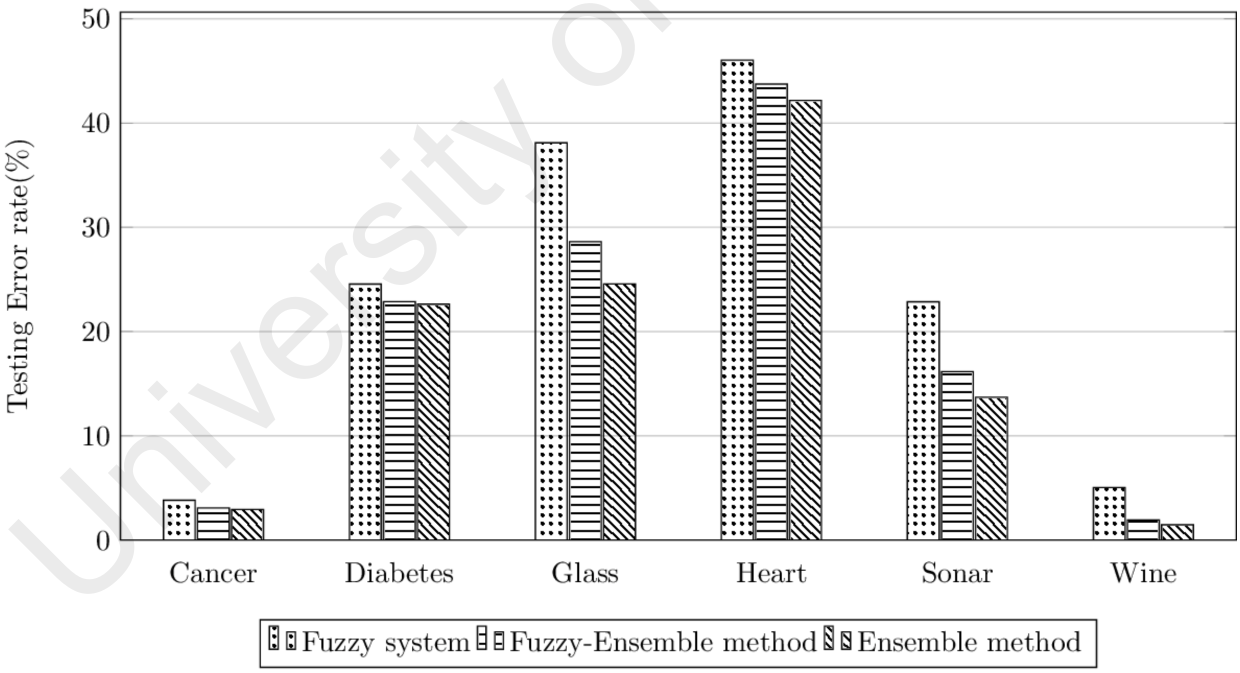


Figure 5.18 Testing error rates (in %) for fuzzy rule-based system, ensemble method and fuzzy-ensemble method

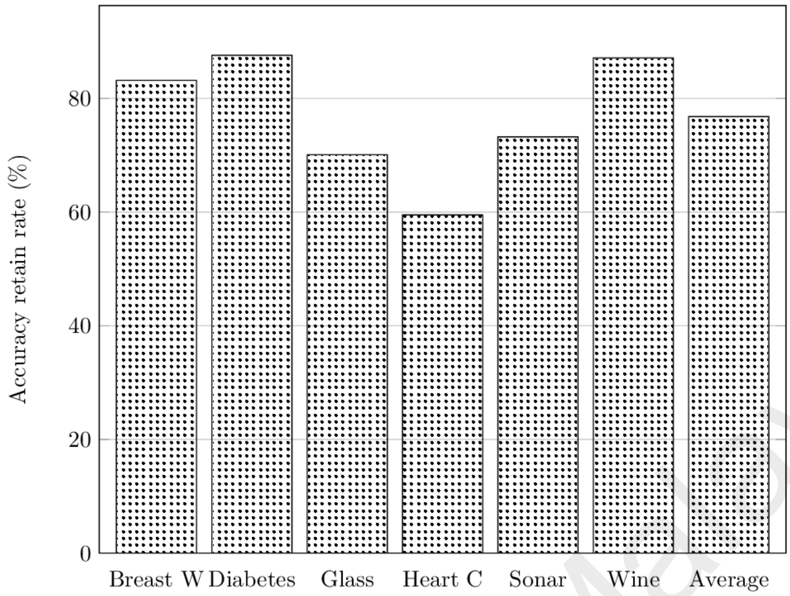


Figure 5.19 Percentage of retained accuracy by fuzzy-ensemble method that was gained by ensemble method relative to fuzzy rule-based system

## 5.4 Summary

This chapter reports the results of the proposed fuzzy-ensemble method that aims to combine the highly accurate ensemble method with the interpretable fuzzy rule-based system. In the first part, we conducted comparisons between the fuzzy rule-based system proposed in (Ishibuchi & Nojima, 2007) which we called Original and two variant methods of the Original algorithm named Proposal1 and Proposal2. The comparisons between the three methods found that Proposal2 achieved the best results and thus it was selected to represent the fuzzy rule-based system in the fuzzy-ensemble method. The second part aims at selecting an ensemble method with the highest classification accuracy. After conducting a series of comparison between different ensemble methods with different settings, we found that Random Linear Oracle (RLO) which combines all the five base classifiers and uses GA-based selection method for choosing a subset of the ensemble outputs achieved the best results in term of accuracy. Because we used two different measures for GA fitness function, the results suggested that using accuracy and double fault measures is better than using accuracy and difficulty measures. Thus, RLO(S-(Acc+DF)), or RLO with GA-based selection method which used accuracy and double default as measures for fitness function, is selected as the ensemble method. The last part was devoted to study the relationship between the ensemble method and the fuzzy rule-based system. We found that fuzzy-ensemble method was able to maintain to a great extend the superiority of the ensemble method accuracy over the fuzzy rule-based system while improves the local interpretability of the fuzzy rule-based system.

# CHAPTER 6: CONCLUSIONS AND FUTURE WORKS

The main objective of this study is to combine the interpretability feature of fuzzy rule-based systems with the accuracy property of the ensemble methods. In this chapter, a summary of the main contributions of this research work is provided in the next section (6.1), while the main conclusions and findings drawn from this research and some suggestions for future works are given in section 6.2 and 6.3, respectively.

## 6.1 Research contributions

The current study contributes to the fields of fuzzy systems and data mining by bringing together two different methods in order to benefit from the strengths of each method. The main contributions of this study can be summarized in the following points:

- Propose a variant method of an efficient evolutionary-fuzzy algorithm proposed in (Ishibuchi & Nojima, 2007) by replacing its multi-objective genetic algorithm called NSGA-II by an enhanced version called Controlled Elitism NSGA-II.

- Propose a feature-based selection method that favors the important features in the fuzzy rules which results in improving the quality of the fuzzy rule-based systems created in the initial population. This guided initialization instead of random one as in (Ishibuchi & Nojima, 2007) has had a positive effect on the performance of the final fuzzy rule-based systems generated.

- Propose a design for an ensemble method that combine five different classifiers and used a GA-based selection method to choose a subset that maximize the ensemble method performance.

- The two developed methods, namely, the fuzzy rule-based system and the ensemble method are standalone methods, i.e. they can be used separately as a fuzzy rule-based system and an ensemble method without the need to be combined. In fact, the comparison of their results with related benchmark methods showed that they are competitive or even better than the existing methods.

- Propose a fuzzy-ensemble method which maintains the interpretability of fuzzy rule-based system while improves its accuracy.

## 6.2 Conclusions

The main conclusions can be summarized in the following points:

- Propsal2 which is a variant method of Original algorithm proposed in (Ishibuchi & Nojima, 2007) was found to be efficient in maximizing the trade-off between the accuracy and interpretability. Even though we selected the fuzzy rule-based system with the highest number of rules and antecedent conditions, among the non-dominated fuzzy systems generated by Proposal2, for combining with the ensemble method; its comparison with other fuzzy systems from the literature has found that it is competitive or even better than these methods in terms of interpretability and accuracy.

- There is a consistent interpretability-accuracy trade-off with respect to error rates on the training data sets for the three methods, namely, Original, Proposal1 and Proposal2, but this trade-off in testing data sets was not consistent. This finding confirms the same conclusion drawn in (Ishibuchi & Nojima, 2007).

- Controlled Elitism NSGA-II algorithm is more efficient than NSGA-II in finding more non-dominated fuzzy rule-based systems with better generalization ability. These findings indicate the potential of Controlled Elitism NSGA-II algorithm compared to NSGA-II.

- Proposal2 has successfully provided high-quality solutions as it has consistently achieved the best error rates for all the six data sets compared to the original method and Proposal1.

- Using feature selection-based approach to select “don’t care” (or less relevant) antecedent conditions in the initial population may give better results compared to the use of random selection method proposed in (Ishibuchi & Nojima, 2007). This may due to the positive effect of the selected fuzzy sets on the quality of the rules generated in the initial population, and the start from a good initial population may help GA to find better solutions compared to a randomly generated initial population.

- Our approach to select the most complicated fuzzy rule-based system among the non-dominated fuzzy rule-based systems seems to be an acceptable one for our case, as the selected fuzzy classifier found to be competitive or even better than the benchmark fuzzy classifiers in both accuracy and interpretability measures.

- Compared to their performance as single classifiers, CART and ANNs as base classifiers for the ensemble methods have made clearly significant gains of accuracy rate (more than 7%) compared with other classifiers (less than 2% for NB and less than 1% for LDA and SVM).

- If we consider the improvements in average ranks of classifiers, we found that ANNs has benefited the most. ANNs whose average ranks as a single classifier was 4 out of 5, became as a base classifier, at the top of the ensemble methods.

- For the ensemble methods, Random Linear Oracle (RLO) achieved the best average ranks. Specifically, RLO achieved the best results in three base classifiers: LDA, ANNs and SVMs while shared the best results with Random Sphere Oracle (RSO) in Naïve Bayes (NB). In addition, Bagging achieved the best results for base classifier CART.

- RLO, Bagging and RSO have successfully enhanced the accuracy of all single classifiers. For Adaboost and Random Subspace, they could enhance the accuracy for all the classifiers except SVMs and LDA. Thus, SVMs and LDA are not recommended for use with AdaBoost or Random Subspace.

- The idea of combining the outputs of ensemble methods with different base classifiers seems better than using an ensemble method with only one base classifier. This may due to the diversity resulted from combining outputs of different base classifiers. In addition, using GA-based selection method with appropriate measure (such as accuracy and double default) is better than considers the outputs of all the ensemble methods (without selection).

- The top 10 ensemble methods, out of 48 in terms of accuracy, are those which combined all the base classifiers for designing the ensemble methods. In addition, the application of GA-based selection method on the outputs of RLO ensemble methods with accuracy and double fault as two measures for the fitness function was found to be the most accurate method.

- The two developed methods, namely, the fuzzy rule-based system and the ensemble method have shown separately competitive results with their related methods. Thus, in addition to the proposed fuzzy-ensemble method, they can be separately used as single classifiers.

- The fuzzy-ensemble method maintains to a great extend the superiority of the ensemble method accuracy over the fuzzy rule-based system by successfully retained an average of 76.77% of the accuracy gains obtained by ensemble method relative to the fuzzy rule-based system. In addition, the fuzzy-ensemble method did not lose its interpretability compared to the fuzzy rule-based system. In contrary, it improved its local interpretability rate.

### 6.3 Limitations and suggestions for future works

We can summarize the possible ways to extend and improve this work in the following points:

- Instead of using Controlled Elitism NSGA-II as a multi-objective genetic algorithm, other multi-objective genetic algorithms proposed in the literature can be used to test their efficiency in maximizing the accuracy-interpretability trade-off in fuzzy rule-based systems.

- We can use different feature selection methods to get the probability of attributes' selection instead of using the seven methods that were chosen for this study.

- Since the proposed fuzzy-ensemble method is a kind of framework, either fuzzy rule-based system or ensemble method can be replaced. That is, Proposal2 which was selected as the fuzzy rule-based system can be replaced with another interpretable fuzzy rule-based system. The same thing can be done for the ensemble method. The only difference lies in the performance of the two classifiers which affect the fuzzy-ensemble classification accuracy.

- We can study the effect of the method developed in Proposal2 on the convergence speed of the multi-objective genetic algorithm and compare it with the original work.

- The proposed method can be applied on more data sets and various real-life applications.

## REFERENCES


Ainon, R. N., Lahsasna, A., & Bulgiba, A. M. (2012). AMI Screening Using Linguistic Fuzzy Rules. *Journal of Medical Systems, 36*, 463–473.

Akbarzadeh, V., Sadeghian, A., & dos Santos, M. V. (2008, 1-6 June 2008). *Derivation of relational fuzzy classification rules using evolutionary computation.* Paper presented at the Fuzzy Systems, 2008. FUZZ-IEEE 2008. (IEEE World Congress on Computational Intelligence). IEEE International Conference on.

Alcalá-Fdez, J., Herrera, F., Márquez, F., & Peregrín, A. (2007). Increasing fuzzy rules cooperation based on evolutionary adaptive inference systems. *International Journal of Intelligent Systems, 22*(9), 1035-1064. doi: 10.1002/int.20237

Alcala, R., Alcala-Fdez, J., Gacto, M. J., & Herrera, F. (2007). Rule base reduction and genetic tuning of fuzzy systems based on the linguistic 3-tuples representation. *Soft Computing, 11*(5), 401-419. doi: DOI 10.1007/s00500-006-0106-2

Alcala, R., Alcala-Fdez, J., & Herrera, F. (2007). A proposal for the genetic lateral tuning of linguistic fuzzy systems and its interaction with rule selection. *Ieee Transactions on Fuzzy Systems, 15*(4), 616-635. doi: Doi 10.1109/Tfuzz.2006.889880

Alcala, R., Ducange, P., Herrera, F., Lazzerini, B., & Marcelloni, F. (2009). A Multiobjective Evolutionary Approach to Concurrently Learn Rule and Data Bases of Linguistic Fuzzy-Rule-Based Systems. *Fuzzy Systems, IEEE Transactions on, 17*(5), 1106-1122. doi: 10.1109/TFUZZ.2009.2023113

Alcala, R., Gacto, M. J., Herrera, F., & Alcala-Fdez, J. (2007). A multi-objective genetic algorithm for tuning and rule selection to obtain accurate and compact linguistic fuzzy rule-based systems. *International Journal of Uncertainty Fuzziness and Knowledge-Based Systems, 15*(5), 539-557. doi: Doi 10.1142/S0218488507004868

Alcalá, R., J., A.-F., Herrera, F., & Otero, J. (2007). Genetic learning of accurate and compact fuzzy rule based systems based on the 2-tuples linguistic representation. *International Journal of Approximate Reasoning, 44*(1), 45-64. doi: http://dx.doi.org/10.1016/j.ijar.2006.02.007

Alonso, J. M. (2007). *Interpretable Fuzzy Systems Modeling With Cooperation Between Expert and Induced Knowledge.* (PhD Thesis), Technical University of Madrid, Spain.

Alonso, J. M., & Magdalena, L. (2010, 18-23 July 2010). *Combining user's preferences and quality criteria into a new index for guiding the design of fuzzy systems with* *a good interpretability-accuracy trade-off.* Paper presented at the Fuzzy Systems (FUZZ), 2010 IEEE International Conference on.

Alonso, J. M., & Magdalena, L. (2011a). HILK++: an interpretability-guided fuzzy modeling methodology for learning readable and comprehensible fuzzy rule-based classifiers. *Soft Computing, 15*(10), 1959-1980. doi: 10.1007/s00500-010-0628-5

Alonso, J. M., & Magdalena, L. (2011b). Special issue on interpretable fuzzy systems. *Information Sciences, 181*(20), 4331-4339. doi: http://dx.doi.org/10.1016/j.ins.2011.07.001

Alonso, J. M., Magdalena, L., & Cordon, O. (2010, 17-19 March 2010). *Embedding HILK in a three-objective evolutionary algorithm with the aim of modeling highly interpretable fuzzy rule-based classifiers.* Paper presented at the Genetic and Evolutionary Fuzzy Systems (GEFS), 2010 4th International Workshop on.

Alonso, J. M., Magdalena, L., & González-Rodríguez, G. (2009). Looking for a good fuzzy system interpretability index: An experimental approach. *International Journal of Approximate Reasoning, 51*(1), 115-134. doi: http://dx.doi.org/10.1016/j.ijar.2009.09.004

Alonso, J. M., Magdalena, L., & Guillaume, S. (2008). HILK: A new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism. *International Journal of Intelligent Systems, 23*(7), 761-794.

Alves, R. T., Delgado, M. R. B., Lopes, H. S., & Freitas, A. (2004). *Induction of fuzzy classification rules with an artificial immune system.* Paper presented at the Proceedings of Brazilian Symposium on Neural Networks, Brazil.

Antonelli, M., Ducange, P., Lazzerini, B., & Marcelloni, F. (2009a). Learning concurrently partition granularities and rule bases of Mamdani fuzzy systems in a multi-objective evolutionary framework. *International Journal of Approximate Reasoning, 50*(7), 1066-1080.

Antonelli, M., Ducange, P., Lazzerini, B., & Marcelloni, F. (2009b). Multi-objective evolutionary learning of granularity, membership function parameters and rules of Mamdani fuzzy systems. *Evolutionary Intelligence, 2*(1), 21-37.

Antonelli, M., Ducange, P., Lazzerini, B., & Marcelloni, F. (2010). *Exploiting a three-objective evolutionary algorithm for generating Mamdani fuzzy rule-based systems.* Paper presented at the Fuzzy Systems (FUZZ), 2010 IEEE International Conference on.

Antonelli, M., Ducange, P., & Marcelloni, F. (2014). A fast and efficient multi-objective evolutionary learning scheme for fuzzy rule-based classifiers. *Information Sciences, 283*, 36-54. doi: http://dx.doi.org/10.1016/j.ins.2014.06.014

Antonelli, M., Frosini, G., Lazzerini, B., & Marcelloni, F. (2006). *A CAD system for lung nodule detection based on an anatomical model and a fuzzy neural network.* Paper presented at the Fuzzy Information Processing Society, 2006. NAFIPS 2006. Annual meeting of the North American.

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., . . . Stoica, I. (2010). A view of cloud computing. *Communications of the ACM, 53*(4), 50-58.

Aymerich, F. X., Alonso, J., Cabañas, M. E., Comabella, M., Sobrevilla, P., & Rovira, A. (2011). Decision tree based fuzzy classifier of magnetic resonance spectra from cerebrospinal fluid samples. *Fuzzy Sets and Systems, 170*(1), 43-63. doi: http://dx.doi.org/10.1016/j.fss.2011.01.003

Babuska, R., & Verbruggen, H. (2003). Neuro-fuzzy methods for nonlinear system identification. *Annual Reviews in Control, 27*(1), 73-85. doi: 10.1016/S1367-5788(03)00009-9

Baesens, B., Setiono, R., Mues, C., & Vanthienen, J. (2003). Using neural network rule extraction and decision tables for credit-risk evaluation. *Management Science, 49*(3), 312-329. doi: 10.1287/mnsc.49.3.312.12739

Bastian, A. (1995). Handling the nonlinearity of a fuzzy logic controller at the transition between rules. *Fuzzy Sets and Systems, 71*(3), 369-387.

Berenji, H. R., & Khedkar, P. (1992). Learning and tuning fuzzy logic controllers through reinforcements. *Neural Networks, IEEE Transactions on, 3*(5), 724-740.

Berlanga, F. J., del Jesus, M. J., Gacto, M. J., & Herrera, F. (2006). A genetic-programming-based approach for the learning of compact fuzzy rule-based classification systems. *Artificial Intelligence and Soft Computing - Icaisc 2006, Proceedings, 4029*, 182-191.

Berlanga, F. J., Rivera, A. J., del Jesus, M. J., & Herrera, F. (2010). GP-COACH: Genetic Programming-based learning of COmpact and ACcurate fuzzy rule-based classification systems for High-dimensional problems. *Information Sciences, 180*(8), 1183-1200. doi: DOI 10.1016/j.ins.2009.12.020

Bezdek, J. C. (1981). *Pattern recognition with fuzzy objective function algorithms*: Kluwer Academic Publishers.

Bodenhofer, U., & Bauer, P. (2003). A formal model of interpretability of linguistic variables. In J. Casillas, O. Cordón, F. Herrera & L. Magdalena (Eds.), *Interpretability Issues in Fuzzy Modelling*. Heidelberg, Germany: Springer-Verlag.

Bonissone, P. P., Khedkar, P. S., & Chen, Y. (1996, 8-11 Sep 1996). *Genetic algorithms for automated tuning of fuzzy controllers: a transportation application.* Paper presented at the Fuzzy Systems, 1996., Proceedings of the Fifth IEEE International Conference on.

Botta, A., Lazzerini, B., & Marcelloni, F. (2008). Context adaptation of Mamdani fuzzy rule based systems. *International Journal of Intelligent Systems, 23*(4), 397-418. doi: Doi 10.1002/Int.20273

Botta, A., Lazzerini, B., Marcelloni, F., & Stefanescu, D. C. (2009). Context adaptation of fuzzy systems through a multi-objective evolutionary approach based on a novel interpretability index. *Soft Computing, 13*(5), 437-449. doi: DOI 10.1007/s00500-008-0360-6

Breiman, L. (1996). Bagging Predictors. *Machine Learning, 24*(2), 123-140. doi: 10.1023/A:1018054314350

Breiman, L. (2001). Random Forests. *Machine Learning, 45*(1), 5-32. doi: 10.1023/A:1010933404324

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*: CRC press.

Brown, G. (2009). *A new perspective for information theoretic feature selection.* Paper presented at the International conference on artificial intelligence and statistics.

Brown, G., Pocock, A., Zhao, M.-J., & Luján, M. (2012). Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. *The Journal of Machine Learning Research, 13*, 27-66.

Brown, M., & Harris, C. (1994). *Neurofuzzy adaptive modelling and control*: Prentice Hall International (UK) Ltd.

Burges, C. C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery, 2*(2), 121-167. doi: 10.1023/A:1009715923555

Cannone, R., Alonso, J., & Magdalena, L. (2011). *Multi-objective design of highly interpretable fuzzy rule-based classifiers with semantic cointension.* Paper presented at the Genetic and Evolutionary Fuzzy Systems (GEFS), 2011 IEEE 5th International Workshop on.

Carse, B., Fogarty, T. C., & Munro, A. (1996). Evolving fuzzy rule based controllers using genetic algorithms. *Fuzzy Sets and Systems, 80*(3), 273-293. doi: Doi 10.1016/0165-0114(95)00196-4

Casillas, J., Cordon, O., del Jesus, M. J., & Herrera, F. (2005). Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction. *Fuzzy Systems, IEEE Transactions on, 13*(1), 13-29.

Casillas, J., Cordón, O., Herrera, F., & Magdalena, L. (2003a). Accuracy improvements to find the balance interpretability-accuracy in linguistic fuzzy modeling: an overview. In J. Casillas, O. Cordón, F. Herrera & L. Magdalena (Eds.), *Accuracy improvements in linguistic fuzzy modelling* (pp. 3-26). Heidelberg, Germany: Springer-Verlag.

Casillas, J., Cordón, O., Herrera, F., & Magdalena, L. (2003b). *Interpretability issues in fuzzy modeling*. Heidelberg, Germany: Springer-Verlag.

Castellano, G., Castiello, C., Fanelli, A., & Mencar, C. (2005). Knowledge discovery by a neuro-fuzzy modeling framework. *Fuzzy Sets and Systems, 149*(1), 187-207.

Castro, P. D., Coelho, G. P., Caetano, M. F., & Zuben, F. J. (2005). Designing Ensembles of Fuzzy Classification Systems: An Immune-Inspired Approach. In C. Jacob, M. L. Pilat, P. J. Bentley & J. I. Timmis (Eds.), *Artificial Immune Systems* (Vol. 3627, pp. 469-482): Springer Berlin Heidelberg.

Chandra, A., & Yao, X. (2006). Evolving hybrid ensembles of learning machines for better generalisation. *Neurocomputing, 69*(7–9), 686-700. doi: http://dx.doi.org/10.1016/j.neucom.2005.12.014

Chen, M.-Y., & Linkens, D. A. (2004). Rule-base self-generation and simplification for data-driven fuzzy models. *Fuzzy Sets and Systems, 142*(2), 243-265. doi: http://dx.doi.org/10.1016/S0165-0114(03)00160-X

Chien, B. C., Lin, J. Y., & Hong, T. P. (2002). Learning discriminant functions with fuzzy attributes for classification using genetic programming. *Expert Systems with Applications, 23*(1), 31-37. doi: http://dx.doi.org/10.1016/S0957-4174(02)00025-8

Chiu, S. L. (1994). Fuzzy model identification based on cluster estimation. *Journal of intelligent and Fuzzy systems, 2*(3), 267-278.

Chow, C. (1970). On optimum recognition error and reject tradeoff. *Information Theory, IEEE Transactions on, 16*(1), 41-46. doi: 10.1109/TIT.1970.1054406

Cococcioni, M., Ducange, P., Lazzerini, B., & Marcelloni, F. (2007). A Pareto-based multi-objective evolutionary approach to the identification of Mamdani fuzzy systems. *Soft Computing-A Fusion of Foundations, Methodologies and Applications, 11*(11), 1013-1031.

Cohen, W. W., & Singer, Y. (1999). *A simple, fast, and effective rule learner.* Paper presented at the Proceedings of the National Conference on Artificial Intelligence.

Cordon, O. (2011). A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. *International Journal of Approximate Reasoning, 52*(6), 894-913. doi: DOI 10.1016/j.ijar.2011.03.004

Cordón, O., Del Jesus, M., Herrera, F., & Lozano, M. (1999). MOGUL: a methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach. *International Journal of Intelligent Systems, 14*(11), 1123-1153.

Cordon, O., del Jesus, M. J., & Herrera, F. (1998). Genetic learning of fuzzy rule-based classification systems cooperating with fuzzy reasoning methods. *International Journal of Intelligent Systems, 13*(10-11), 1025-1053. doi: Doi 10.1002/(Sici)1098-111x(199810/11)13:10/11<1025::Aid-Int9>3.3.Co;2-O

Cordón, O., del Jesus, M. J., & Herrera, F. (1999). A proposal on reasoning methods in fuzzy rule-based classification systems. *International Journal of Approximate Reasoning, 20*(1), 21-45.

Cordon, O., Del Jesus, M. J., Herrera, F., Magdalena, L., & Villar, P. (2003). A multiobjective genetic learning process for joint feature selection and granularity and contexts learning in fuzzy rule-based classification systems. In J. Casillas, O. Cordón, F. Herrera & L. Magdalena (Eds.), *Interpretability Issues in Fuzzy Modeling* (Vol. 128, pp. 79-99). Heidelberg, Germany: Springer-Verlag.

Cordon, O., & Herrera, F. (1997). A three-stage evolutionary process for learning descriptive and approximate fuzzy-logic-controller knowledge bases from examples. *International Journal of Approximate Reasoning, 17*(4), 369-407. doi: Doi 10.1016/S0888-613x(96)00133-8

Cordón, O., Herrera, F., Del Jesus, M., & Villar, P. (2001). *A multiobjective genetic algorithm for feature selection and granularity learning in fuzzy-rule based classification systems.* Paper presented at the IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th.

Cordón, O., Herrera, F., Magdalena, L., & Villar, P. (2001). A genetic learning process for the scaling factors, granularity and contexts of the fuzzy rule-based system data base. *Information Sciences, 136*(1–4), 85-107. doi: http://dx.doi.org/10.1016/S0020-0255(01)00143-8

Cordon, O., Herrera, F., & Villar, P. (2001). Generating the knowledge base of a fuzzy rule-based system by the genetic learning of the data base. *Fuzzy Systems, IEEE Transactions on, 9*(4), 667-674. doi: 10.1109/91.940977

Cordon, O., Quirin, A., & Sanchez, L. (2008a, 4-7 March 2008). *A first study on bagging fuzzy rule-based classification systems with multicriteria genetic selection of the component classifiers.* Paper presented at the Genetic and Evolving Systems, 2008. GEFS 2008. 3rd International Workshop on.

Cordon, O., Quirin, A., & Sanchez, L. (2008b, 10-12 Sept. 2008). *On the Use of Bagging, Mutual Information-Based Feature Selection and Multicriteria Genetic Algorithms to Design Fuzzy Rule-Based Classification Ensembles.* Paper presented at the Hybrid Intelligent Systems, 2008. HIS '08. Eighth International Conference on.

Crockett, K., Bandar, Z., & Mclean, D. (2007). *On the optimization of T-norm parameters within fuzzy decision trees.* Paper presented at the Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International.

Crockett, K., Bandar, Z., McLean, D., & O'Shea, J. (2006). On constructing a fuzzy inference framework using crisp decision trees. *Fuzzy Sets and Systems, 157*(21), 2809-2832. doi: http://dx.doi.org/10.1016/j.fss.2006.06.002

Das, S. (2001). *Filters, Wrappers and a Boosting-Based Hybrid for Feature Selection*. Paper presented at the Proceedings of the Eighteenth International Conference on Machine Learning.

De Oliveira, J. V. (1999). Semantic constraints for membership function optimization. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 29*(1), 128-138.

Deb, K., & Goel, T. (2001). Controlled Elitist Non-dominated Sorting Genetic Algorithms for Better Convergence. In E. Zitzler, L. Thiele, K. Deb, C. Coello Coello & D. Corne (Eds.), *Evolutionary Multi-Criterion Optimization* (Vol. 1993, pp. 67-81): Springer Berlin Heidelberg.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on, 6*(2), 182-197. doi: 10.1109/4235.996017

Demuth, H., & Beale, M. (2010). *Neural network toolbox user's guide*. 24 Prime Park Way, Natick, MA 01760-1500: The Math Works.

Detrano, R., Janosi, A., Steinbrunn, W., Pfisterer, M., Schmid, J.-J., Sandhu, S., . . . Froelicher, V. (1989). International application of a new probability algorithm for the diagnosis of coronary artery disease. *The American journal of cardiology, 64*(5), 304-310.

Devijver, P. A., & Kittler, J. (1982). *Pattern recognition: A statistical approach*: Prentice/Hall International Englewood Cliffs, NJ.

Dietterich, T. G. (2000). Ensemble methods in machine learning. *Multiple Classifier Systems, 1857*, 1-15.

Domingos, P. (1998). Knowledge discovery via multiple models. *Intelligent Data Analysis, 2*(1–4), 187-202. doi: http://dx.doi.org/10.1016/S1088-467X(98)00023-7

Dubois, D., & Prade, H. (1980). *Fuzzy sets and systems: theory and applications* (Vol. 144): Academic press.

Dubois, D., & Prade, H. (1996). What are fuzzy rules and how to use them. *Fuzzy Sets and Systems, 84*(2), 169-185. doi: http://dx.doi.org/10.1016/0165-0114(96)00066-8

Dubois, D., Prade, H., & Ughetto, L. (1997). Checking the coherence and redundancy of fuzzy knowledge bases. *Fuzzy Systems, IEEE Transactions on, 5*(3), 398-417. doi: 10.1109/91.618276

Ducange, P., Lazzerini, B., & Marcelloni, F. (2010a). Multi-objective genetic fuzzy classifiers for imbalanced and cost-sensitive datasets. *Soft Computing, 14*(7), 713-728. doi: 10.1007/s00500-009-0460-y

Ducange, P., Lazzerini, B., & Marcelloni, F. (2010b). Multi-objective genetic fuzzy classifiers for imbalanced and cost-sensitive datasets. *Soft Computing-A Fusion of Foundations, Methodologies and Applications, 14*(7), 713-728.

Dunn, J. C. (1973). A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters.

Elomaa, T., & Rousu, J. (1999). General and Efficient Multisplitting of Numerical Attributes. *Machine Learning, 36*(3), 201-244. doi: 10.1023/A:1007674919412

Elovici, Y., Shapira, B., & Kantor, P. B. (2006). A decision theoretic approach to combining information filters: An analytical and empirical evaluation. *Journal of the American Society for Information Science and Technology, 57*(3), 306-320.

Espinosa, J., & Vandewalle, J. (2000). Constructing fuzzy models with linguistic integrity from numerical data-AFRELI algorithm. *Fuzzy Systems, IEEE Transactions on, 8*(5). doi: 10.1109/91.873582

Fazendeiro, P., & de Oliveira, J. V. (2005). *A working hypothesis on the semantics/accuracy synergy*. Paper presented at the Joint EUSFLAT-LFA, Barcelona, Spain.

Fazzolari, M., Alcalá, R., & Herrera, F. (2014). A multi-objective evolutionary method for learning granularities based on fuzzy discretization to improve the accuracy-complexity trade-off of fuzzy rule-based classification systems: D-MOFARC algorithm. *Applied Soft Computing, 24*, 470-481. doi: http://dx.doi.org/10.1016/j.asoc.2014.07.019

Fazzolari, M., Alcala, R., Nojima, Y., Ishibuchi, H., & Herrera, F. (2013). A Review of the Application of Multiobjective Evolutionary Fuzzy Systems: Current Status and Further Directions. *Fuzzy Systems, IEEE Transactions on, 21*(1), 45-65. doi: 10.1109/TFUZZ.2012.2201338

Feldman, D. S. (1993). *Fuzzy network synthesis with genetic algorithms*. Paper presented at the Proceedings of the 5th international conference on genetic algorithms, Univ. Illinois

Fernandez, A., del Jesus, M. J., & Herrera, F. (2010). On the 2-tuples based genetic tuning performance for fuzzy rule based classification systems in imbalanced data-sets. *Information Sciences, 180*(8), 1268-1291. doi: DOI 10.1016/j.ins.2009.12.014

Fernández, A., López, V., del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems, 80*, 109-121. doi: http://dx.doi.org/10.1016/j.knosys.2015.01.013

Ferri, C., Hernández-Orallo, J., & Ramírez-Quintana, M. J. (2002). From Ensemble Methods to Comprehensible Models. In S. Lange, K. Satoh & C. Smith (Eds.), *Discovery Science* (Vol. 2534, pp. 165-177): Springer Berlin Heidelberg.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics, 7*(2), 179-188. doi: 10.1111/j.1469-1809.1936.tb02137.x

Fleuret, F. (2004). Fast binary feature selection with conditional mutual information. *The Journal of Machine Learning Research, 5*, 1531-1555.

Fonseca, C. M., & Fleming, P. J. (1993). *Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization.* Paper presented at the Proceedings of the fifth international conference on genetic algorithms.

Foukal, P. V. (1990). Changeable sun. *Scientific American, 262*(2), 8-15.

Freund, Y., & Mason, L. (1999). *The alternating decision tree learning algorithm.* Paper presented at the ICML.

Freund, Y., & Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences, 55*(1), 119-139. doi: http://dx.doi.org/10.1006/jcss.1997.1504

Fumera, G. (2002). *Advanced Methods for Pattern Recognition with Reject Option.* (Ph. D), University of Cagliari, Italy

Fumera, G., Roli, F., & Giacinto, G. (2000). Reject option with multiple thresholds. *Pattern Recognition, 33*(12), 2099-2101. doi: http://dx.doi.org/10.1016/S0031-3203(00)00059-5

Furuhashi, T., Nakaoka, K., & Uchikawa, Y. (1994, 26-29 Jun 1994). *Suppression of excessive fuzziness using multiple fuzzy classifier systems.* Paper presented at the Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on.

Gacto, M. J., Alcala, R., & Herrera, F. (2010). Integration of an Index to Preserve the Semantic Interpretability in the Multiobjective Evolutionary Rule Selection and Tuning of Linguistic Fuzzy Systems. *Fuzzy Systems, IEEE Transactions on, 18*(3), 515-531. doi: 10.1109/TFUZZ.2010.2041008

Gacto, M. J., Alcalá, R., & Herrera, F. (2009). Adaptation and application of multi-objective evolutionary algorithms for rule reduction and parameter tuning of fuzzy rule-based systems. *Soft Computing-A Fusion of Foundations, Methodologies and Applications, 13*(5), 419-436.

Gacto, M. J., Alcalá, R., & Herrera, F. (2011). Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. *Information Sciences, 181*(20), 4340-4360. doi: http://dx.doi.org/10.1016/j.ins.2011.02.021

Gacto, M. J., Alcalá, R., & Herrera, F. (2012). A multi-objective evolutionary algorithm for an effective tuning of fuzzy logic controllers in heating, ventilating and air conditioning systems. *Applied Intelligence, 36*(2), 330-347. doi: 10.1007/s10489-010-0264-x

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning, 63*(1), 3-42.

Giacinto, G., & Roli, F. (2001). Design of effective neural network ensembles for image classification purposes. *Image and Vision Computing, 19*(9-10), 699-707. doi: 10.1016/s0262-8856(01)00045-2

Glorennec, P. Y. (1997). Coordination between autonomous robots. *International Journal of Approximate Reasoning, 17*(4), 433-446. doi: http://dx.doi.org/10.1016/S0888-613X(97)00004-2

Gomes, L. C. T., de Sousa, J. S., Bezerra, G. B., de Castro, L. N., & Von Zuben, F. J. (2003). *Copt-aiNet and the Gene Ordering Problem.* Paper presented at the Second Brazilian Workshop on Bioinformatics, Macae, Brazil

Gonzalez, A., & Perez, R. (1999). SLAVE: A genetic learning system based on an iterative approach. *Ieee Transactions on Fuzzy Systems, 7*(2), 176-191. doi: Doi 10.1109/91.755399

González, A., & Pérez, R. (1999). A study about the inclusion of linguistic hedges in a fuzzy rule learning algorithm. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 7*(03), 257-266.

Grosan, C., & Abraham, A. (2007). Hybrid Evolutionary Algorithms: Methodologies, Architectures, and Reviews. In A. Abraham, C. Grosan & H. Ishibuchi (Eds.), *Hybrid Evolutionary Algorithms* (Vol. 75, pp. 1-17): Springer Berlin Heidelberg.

Guillaume, S. (2001). Designing fuzzy inference systems from data: an interpretability-oriented review. *Fuzzy Systems, IEEE Transactions on, 9*(3), 426-443.

Guillaume, S., & Charnomordic, B. (2003). A new method for inducing a set of interpretable fuzzy partitions and fuzzy inference systems from data. In J. Casillas, O. Cordón, F. Herrera & L. Magdalena (Eds.), *Interpretability Issues in Fuzzy Modelling* (pp. 148-175). Heidelberg, Germany Springer-Verlag.

Guillaume, S., & Charnomordic, B. (2004). Generating an interpretable family of fuzzy partitions from data. *Fuzzy Systems, IEEE Transactions on, 12*(3), 324-335. doi: 10.1109/TFUZZ.2004.825979

Guillaume, S., & Charnomordic, B. (2011). Learning interpretable fuzzy inference systems with FisPro. *Information Sciences*, *181*(20), 4409-4427. doi: http://dx.doi.org/10.1016/j.ins.2011.03.025

Guillaume, S., & Charnomordic, B. (2012). Fuzzy inference systems: An integrated modeling environment for collaboration between expert knowledge and data using FisPro. *Expert Systems with Applications, 39*(10), 8744-8755. doi: http://dx.doi.org/10.1016/j.eswa.2012.01.206

Gürocak, H. B. (1999). A genetic-algorithm-based method for tuning fuzzy logic controllers. *Fuzzy Sets and Systems*, *108*(1), 39-47. doi: http://dx.doi.org/10.1016/S0165-0114(97)00309-6

Hamada, Y., Nojima, Y., & Ishibuchi, H. (2009). Use of multi-objective genetic rule selection for examining the effectiveness of inter-vehicle communication in traffic simulations. *Artificial Life and Robotics, 14*(3), 410-413. doi: 10.1007/s10015-009-0697-7

Hand, D. J., & Yu, K. (2001). Idiot's Bayes: Not So Stupid after All? *International Statistical Review / Revue Internationale de Statistique, 69*(3), 385-398. doi: 10.2307/1403452

Hanebeck, U. D., & Schmidt, G. K. (1996). Genetic optimization of fuzzy networks. *Fuzzy Sets and Systems, 79*(1), 59-68.

Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 12*(10), 993-1001. doi: 10.1109/34.58871

Haykin, S. (1998). *Neural networks: a comprehensive foundation*. NJ: Prentice Hall Englewood Cliffs.

Herrera, F. (2008). Genetic fuzzy systems: taxonomy, current research trends and prospects. *Evolutionary Intelligence*, *1*(1), 27-46. doi: 10.1007/s12065-007-0001-5

Herrera, F., Lozano, M., & Verdegay, J. L. (1995). Tuning fuzzy logic controllers by genetic algorithms. *International Journal of Approximate Reasoning, 12*(3–4), 299-315. doi: http://dx.doi.org/10.1016/0888-613X(94)00033-Y

Herrera, F., Lozano, M., & Verdegay, J. L. (1998). A learning process for fuzzy control rules using genetic algorithms. *Fuzzy Sets and Systems, 100*(1-3), 143-158. doi: Doi 10.1016/S0165-0114(97)00043-2

Herrera, F., & Martinez, L. (2000). A 2-tuple fuzzy linguistic representation model for computing with words. *Fuzzy Systems, IEEE Transactions on, 8*(6), 746-752. doi: 10.1109/91.890332

Herrmann, C. S. (1997). Symbolical reasoning about numerical data: A hybrid approach. *Applied Intelligence, 7*(4), 339-354.

Hoffmann, F., Baesens, B., Martens, J., Put, F., & Vanthienen, J. (2002). Comparing a genetic fuzzy and a neurofuzzy classifier for credit scoring. *International Journal of Intelligent Systems, 17*(11), 1067-1083. doi: 10.1002/int.10052

Hoffmann, F., Baesens, B., Mues, C., Van Gestel, T., & Vanthienen, J. (2007). Inferring descriptive and approximate fuzzy rules for credit scoring using evolutionary algorithms. *European Journal of Operational Research, 177*(1), 540-555.

Homaifar, A., & Mccormick, E. (1995). Simultaneous Design of Membership Functions and Rule Sets for Fuzzy Controllers Using Genetic Algorithms. *Ieee Transactions on Fuzzy Systems, 3*(2), 129-139. doi: Doi 10.1109/91.388168

Horn, J., Nafpliotis, N., & Goldberg, D. E. (1994). *A niched Pareto genetic algorithm for multiobjective optimization.* Paper presented at the Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on.

Hühn, J., & Hüllermeier, E. (2009). FURIA: an algorithm for unordered fuzzy rule induction. *Data Mining and Knowledge Discovery, 19*(3), 293-319. doi: 10.1007/s10618-009-0131-8

Hullermeier, E., & Vanderlooy, S. (2009). Why fuzzy decision trees are good rankers. *Fuzzy Systems, IEEE Transactions on, 17*(6), 1233-1244.

Ishibuchi, H. (2007, 23-26 July 2007). *Multiobjective Genetic Fuzzy Systems: Review and Future Research Directions.* Paper presented at the Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International.

Ishibuchi, H., & Murata, T. (1996). A genetic-algorithm-based fuzzy partition method for pattern classification problems. *Genetic algorithms and soft computing*(8), 555-578.

Ishibuchi, H., Murata, T., & Turksen, I. B. (1997). Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems. *Fuzzy Sets and Systems, 89*(2), 135-150. doi: Doi 10.1016/S0165-0114(96)00098-X

Ishibuchi, H., Nakashima, T., & Murata, T. (1999). Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems. *Ieee Transactions on Systems Man and Cybernetics Part B-Cybernetics, 29*(5), 601-618. doi: Doi 10.1109/3477.790443

Ishibuchi, H., Nakashima, T., & Murata, T. (2001). Three-objective genetics-based machine learning for linguistic rule extraction. *Information Sciences, 136*(1-4), 109-133. doi: Doi 10.1016/S0020-0255(01)00144-X

Ishibuchi, H., Nakashima, T., & Nii, M. (2005). *Classification and modeling with linguistic information granules: advanced approaches advanced approaches to linguistic data mining*: Springer.

Ishibuchi, H., & Nakshima, T. (1998, 4-9 May 1998). *Fuzzy classification with reject options by fuzzy if-then rules.* Paper presented at the Fuzzy Systems Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on.

Ishibuchi, H., & Nii, M. (2000). Neural networks for soft decision making. *Fuzzy Sets and Systems, 115*(1), 121-140. doi: http://dx.doi.org/10.1016/S0165-0114(99)00022-6

Ishibuchi, H., & Nojima, Y. (2005, 6-9 Nov. 2005). *Performance evaluation of evolutionary multiobjective approaches to the design of fuzzy rule-based ensemble classifiers.* Paper presented at the Hybrid Intelligent Systems (HIS '05) Fifth International Conference on.

Ishibuchi, H., & Nojima, Y. (2006). Evolutionary multiobjective optimization for the design of fuzzy rule-based ensemble classifiers. *International Journal of Hybrid Intelligent Systems, 3*(3), 129-145.

Ishibuchi, H., & Nojima, Y. (2007). Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning. *International Journal of Approximate Reasoning, 44*(1), 4-31. doi: DOI 10.1016/j.ijar.2006.01.004

Ishibuchi, H., & Nojima, Y. (2013). Repeated double cross-validation for choosing a single solution in evolutionary multi-objective fuzzy classifier design. *Knowledge-Based Systems, 54,* 22–31.

Ishibuchi, H., Nozaki, K., & Tanaka, H. (1992). Distributed Representation of Fuzzy Rules and Its Application to Pattern-Classification. *Fuzzy Sets and Systems, 52*(1), 21-32. doi: Doi 10.1016/0165-0114(92)90032-Y

Ishibuchi, H., Nozaki, K., Tanaka, H., Hosaka, Y., & Matsuda, M. (1994). Empirical study on learning in fuzzy systems by rice taste analysis. *Fuzzy Sets and Systems, 64*(2), 129-144. doi: http://dx.doi.org/10.1016/0165-0114(94)90329-8

Ishibuchi, H., Nozaki, K., Yamamoto, N., & Tanaka, H. (1995). Selecting Fuzzy If-Then Rules for Classification Problems Using Genetic Algorithms. *Ieee Transactions on Fuzzy Systems, 3*(3), 260-270. doi: Doi 10.1109/91.413232

Ishibuchi, H., & Yamamoto, T. (2003). Evolutionary Multiobjective Optimization for Generating an Ensemble of Fuzzy Rule-Based Classifiers. In E. Cantú-Paz, J. A. Foster, K. Deb, L. D. Davis, R. Roy, U.-M. O'Reilly, H.-G. Beyer, R. Standish, G. Kendall, S. Wilson, M. Harman, J. Wegener, D. Dasgupta, M. A. Potter, A. C. Schultz, K. A. Dowsland, N. Jonoska & J. Miller (Eds.), *Genetic and Evolutionary Computation — GECCO 2003* (Vol. 2723, pp. 1077-1088): Springer Berlin Heidelberg.

Ishibuchi, H., & Yamamoto, T. (2004). Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. *Fuzzy Sets and Systems, 141*(1), 59-88. doi: Doi 10.1016/S0165-0114(03)00114-3

Ishibuchi, H., & Yamamoto, T. (2005). Rule weight specification in fuzzy rule-based classification systems. *Ieee Transactions on Fuzzy Systems, 13*(4), 428-435. doi: Doi 10.1109/Tfuzz.2004.841738

Ishibuchi, H., Yamamoto, T., & Nakashima, T. (2005). Hybridization of fuzzy GBML approaches for pattern classification problems. *Ieee Transactions on Systems Man and Cybernetics Part B-Cybernetics, 35*(2), 359-365. doi: Doi 10.1109/Tsmcb.2004.842257

Ishigami, H., Fukuda, T., Shibata, T., & Arai, F. (1995). Structure Optimization of Fuzzy Neural-Network by Genetic Algorithm. *Fuzzy Sets and Systems, 71*(3), 257-264. doi: Doi 10.1016/0165-0114(94)00283-D

Jakulin, A. (2005). *Machine learning based on attribute interactions.* Univerza v Ljubljani.

Jamei, M., Mahfouf, M., & Linkens, D. A. (2001). *Elicitation and fine-tuning of Mamdani-type fuzzy rules using symbiotic evolution.* Paper presented at the Proceedings of European Symposium on Intelligent Technologies, Hybrid Systems and their Implementation on Smart Adaptive Systems (EUNITE 2001), Tenerife, Spain.

Jang, J. S. R. (1993). ANFIS: adaptive-network-based fuzzy inference system. *Systems, Man and Cybernetics, IEEE Transactions on, 23*(3), 665-685. doi: 10.1109/21.256541

Jang, J. S. R., Sun, C. T., & Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*: PTR Prentice Hall.

Janikow, C. Z. (1996). A genetic algorithm method for optimizing fuzzy decision trees. *Information Sciences, 89*(3-4), 275-296. doi: Doi 10.1016/0020-0255(95)00239-1

Janikow, C. Z. (1998). Fuzzy decision trees: issues and methods. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 28*(1), 1-14.

Jiménez, F., Gómez-Skarmeta, A., Roubos, H., & Babuška, R. (2001). Accurate, Transparent, and Compact Fuzzy Models for Function Approximation and Dynamic Modeling through Multi-objective Evolutionary Optimization. In E. Zitzler, L. Thiele, K. Deb, C. Coello Coello & D. Corne (Eds.), *Evolutionary Multi-Criterion Optimization* (Vol. 1993, pp. 653-667): Springer Berlin Heidelberg.

Jimenez, F., Gomez-Skarmeta, A. F., Roubos, H., & Babuska, R. (2001, 25-28 July 2001). *A multi-objective evolutionary algorithm for fuzzy modeling.* Paper presented at the IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th.

Jin, Y., & Sendhoff, B. (2003). Extracting Interpretable Fuzzy Rules from RBF Networks. *Neural Process. Lett., 17*(2), 149-164. doi: 10.1023/a:1023642126478

Jin, Y., von Seelen, W., & Sendhoff, B. (1999). On generating FC<sup>3</sup> fuzzy rule systems from data using evolution strategies. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 29*(6), 829-845. doi: 10.1109/3477.809036

Juang, C. F., & Lin, C. T. (1998). An online self-constructing neural fuzzy inference network and its applications. *Fuzzy Systems, IEEE Transactions on, 6*(1), 12-32. doi: 10.1109/91.660805

Juang, C. F., Lin, J. Y., & Lin, C. T. (2000). Genetic reinforcement learning through symbiotic evolution for fuzzy controller design. *Ieee Transactions on Systems Man and Cybernetics Part B-Cybernetics, 30*(2), 290-302.

Kandel, A. (1992). *Fuzzy expert systems*: CRC PressI Llc.

Karr, C. (1991). Genetic algorithms for fuzzy controllers. *AI EXPERT., 6*(2), 26-33.

Karr, C. L., & Gentry, E. J. (1993). Fuzzy control of pH using genetic algorithms. *Fuzzy Systems, IEEE Transactions on, 1*(1), 46. doi: 10.1109/TFUZZ.1993.390283

Kasabov, N., & Woodford, B. (1999, 22-25 Aug. 1999). *Rule insertion and rule extraction from evolving fuzzy neural networks: algorithms and applications for building adaptive, intelligent expert systems.* Paper presented at the Fuzzy Systems Conference Proceedings, 1999. FUZZ-IEEE '99. 1999 IEEE International.

Kaufmann, M., Meier, A., & Stoffel, K. (2015). IFC-Filter: Membership function generation for inductive fuzzy classification. *Expert Systems with Applications, 42*(21), 8369-8379. doi: http://dx.doi.org/10.1016/j.eswa.2015.06.034

Keller, J. M., & Hunt, D. J. (1985). Incorporating fuzzy membership functions into the perceptron algorithm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*(6), 693-699.

Keller, J. M., & Tahani, H. (1992). Implementation of conjunctive and disjunctive fuzzy logic rules with neural networks. *International Journal of Approximate Reasoning, 6*(2), 221-240.

Klose, A., Nürnberger, A., & Nauck, D. (1998). *Some approaches to improve the interpretability of neuro-fuzzy classifiers.* Paper presented at the In: Proc. 6th European Congress on Intelligent Techniques and Soft Computing (EUFIT'98).

Klösgen, W. (1996). Explora: a multipattern and multistrategy discovery assistant. In U. Fayyad, G. Piatetsky-Shapiro, P. Smyth & R. Uthurusamy (Eds.), *Advances in knowledge discovery and data mining* (pp. 249-271): American Association for Artificial Intelligence.

Knowles, J. D., & Corne, D. W. (2000). Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary computation, 8*(2), 149-172.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence, 97*(1–2), 273-324. doi: http://dx.doi.org/10.1016/S0004-3702(97)00043-X

Konak, A., Coit, D. W., & Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety, 91*(9), 992-1007. doi: DOI 10.1016/j.ress.2005.11.018

Krone, A., & Taeger, H. (2001). Data-based fuzzy rule test for fuzzy modelling. *Fuzzy Sets and Systems, 123*(3), 343-358. doi: http://dx.doi.org/10.1016/S0165-0114(00)00112-3

Krzysztof, T., Quirin, A., & Cordon, O. (2009). *On the Combination of Accuracy and Diversity Measures for Genetic Selection of Bagging Fuzzy Rule-Based Multiclassification Systems*.

Kuncheva, L. (2004). *Combining Pattern Classifiers* John Wiley & Sons.

Kuncheva, L. (2005). Diversity in multiple classifier systems. *Information fusion, 6*(1), 3-4.

Kuncheva, L., & Rodriguez, J. J. (2007). Classifier Ensembles with a Random Linear Oracle. *Knowledge and Data Engineering, IEEE Transactions on, 19*(4), 500-508. doi: 10.1109/TKDE.2007.1016

Kuncheva, L., & Whitaker, C. (2003). Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy. *Machine Learning, 51*(2), 181-207. doi: 10.1023/A:1022859003006

Kuncheva, L. I. (2007). *A stability index for feature selection.* Paper presented at the Proceedings of the 25th conference on Proceedings of the 25th IASTED International Multi-Conference: artificial intelligence and applications.

Kuncheva, L. I. (2014). *Combining pattern classifiers: methods and algorithms, 2nd Edition*: John Wiley & Sons.

Kuncheva, L. I., Bezdek, J. C., & Duin, R. P. W. (2001). Decision templates for multiple classifier fusion: an experimental comparison. *Pattern Recognition, 34*(2), 299-314. doi: http://dx.doi.org/10.1016/S0031-3203(99)00223-X

Kuncheva, L. I., & Rodriguez, J. J. (2007). Classifier Ensembles with a Random Linear Oracle. *Knowledge and Data Engineering, IEEE Transactions on, 19*(4), 500-508. doi: 10.1109/TKDE.2007.1016

Kuncheva, L. I., & Rodríguez, J. J. (2010). Classifier ensembles for fMRI data analysis: an experiment. *Magnetic Resonance Imaging, 28*(4), 583-593. doi: http://dx.doi.org/10.1016/j.mri.2009.12.021

Lahsasna, A., Ainon, R. N., Roziati, Z., & Bulgiba, A. M. (2012). Design of a Fuzzy-based Decision Support System for Coronary Heart Disease Diagnosis. *Journal of Medical Systems, 36*, 3293–3306.

Lai, C., Reinders, M. J. T., & Wessels, L. (2006). Random subspace method for multivariate feature selection. *Pattern Recognition Letters, 27*(10), 1067-1076. doi: http://dx.doi.org/10.1016/j.patrec.2005.12.018

Lee, C. C. (1990). Fuzzy logic in control systems: fuzzy logic controller. I. *Systems, Man and Cybernetics, IEEE Transactions on*, *20*(2), 404-418.

Leigh, W., Purvis, R., & Ragusa, J. M. (2002). Forecasting the NYSE composite index with technical analysis, pattern recognizer, neural network, and genetic algorithm: a case study in romantic decision support. *Decision Support Systems*, *32*(4), 361-377. doi: http://dx.doi.org/10.1016/S0167-9236(01)00121-X

Lewis, D. D. (1992). *Feature selection and feature extraction for text categorization*. Paper presented at the Proceedings of the workshop on Speech and Natural Language, Harriman, New York.

Liao, S., Chu, P., & Hsiao, P. (2012). Data mining techniques and applications – A decade review from 2000 to 2011. *Expert Systems with Applications*, *39*(12), 11303-11311. doi: http://dx.doi.org/10.1016/j.eswa.2012.02.063

Lin, C. T., & Lee, C. S. G. (1991). Neural-network-based fuzzy logic control and decision system. *Computers, IEEE Transactions on*, *40*(12), 1320-1336. doi: 10.1109/12.106218

Lin, D., & Tang, X. (2006). Conditional Infomax Learning: An Integrated Framework for Feature Extraction and Fusion. In A. Leonardis, H. Bischof & A. Pinz (Eds.), *Computer Vision – ECCV 2006* (Vol. 3951, pp. 68-82): Springer Berlin Heidelberg.

Liu, C.-L. (2005). Classifier combination based on confidence transformation. *Pattern Recognition*, *38*(1), 11-28.

Liu, F., Quek, C., & Ng, G. S. (2007). A Novel Generic Hebbian Ordering-Based Fuzzy Rule Base Reduction Approach to Mamdani Neuro-Fuzzy System. *Neural Computation*, *19*(6), 1656-1680. doi: 10.1162/neco.2007.19.6.1656

Loh, W.-Y. (2011). Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *1*(1), 14-23. doi: 10.1002/widm.8

Louisa, L., & Suen, C. Y. (1997). Application of majority voting to pattern recognition: an analysis of its behavior and performance. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, *27*(5), 553-568. doi: 10.1109/3468.618255

Magdalena, L., & Monasterio-Huelin, F. (1997). A fuzzy logic controller with learning through the evolution of its knowledge base. *International Journal of Approximate Reasoning*, *16*(3), 335-358.

Malhotra, R., & Malhotra, D. K. (2002). Differentiating between good credits and bad credits using neuro-fuzzy systems. *European Journal of Operational Research, 136*(1), 190-211. doi: http://dx.doi.org/10.1016/S0377-2217(01)00052-2

Mamdani, E. H., & Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International journal of man-machine studies, 7*(1), 1-13.

Mangiameli, P., West, D., & Rampal, R. (2004). Model selection for medical diagnosis decision support systems. *Decision Support Systems, 36*(3), 247-259.

Mansoori, E. G., Zolghadri, M. J., & Katebi, S. D. (2008). SGERD: A steady-state genetic algorithm for extracting fuzzy classification rules from data. *Fuzzy Systems, IEEE Transactions on, 16*(4), 1061-1071.

Marquez, A. A., Marquez, F. A., & Peregrin, A. (2010, 18-23 July 2010). *A multi-objective evolutionary algorithm with an interpretability improvement mechanism for linguistic fuzzy systems with adaptive defuzzification.* Paper presented at the Fuzzy Systems (FUZZ), 2010 IEEE International Conference on.

Meesad, P., & Yen, G. G. (2002, 2002). *Quantitative measures of the accuracy, comprehensibility, and completeness of a fuzzy expert system.* Paper presented at the Fuzzy Systems, 2002. FUZZ-IEEE'02. Proceedings of the 2002 IEEE International Conference on.

Mencar, C., Castellano, G., & Fanelli, A. M. (2007). Distinguishability quantification of fuzzy sets. *Information Sciences, 177*(1), 130-149. doi: http://dx.doi.org/10.1016/j.ins.2006.04.008

Mencar, C., Castiello, C., Cannone, R., & Fanelli, A. M. (2011). Interpretability assessment of fuzzy knowledge bases: A cointension based approach. *International Journal of Approximate Reasoning, 52*(4), 501-518. doi: DOI 10.1016/j.ijar.2010.11.007

Mencar, C., & Fanelli, A. M. (2008). Interpretability constraints for fuzzy information granulation. *Information Sciences, 178*(24), 4585-4618. doi: DOI 10.1016/j.ins.2008.08.015

Meyer, P. E., & Bontempi, G. (2006). On the Use of Variable Complementarity for Feature Selection in Cancer Classification. In F. Rothlauf, J. Branke, S. Cagnoni, E. Costa, C. Cotta, R. Drechsler, E. Lutton, P. Machado, J. H. Moore, J. Romero, G. D. Smith, G. Squillero & H. Takagi (Eds.), *Applications of Evolutionary Computing* (Vol. 3907, pp. 91-102): Springer Berlin Heidelberg.

Mikut, R., Jäkel, J., & Gröll, L. (2005). Interpretability issues in data-based learning of fuzzy systems. *Fuzzy Sets and Systems*, *150*(2), 179-197.

Miller, G. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *The psychological review*, *63*, 81-97.

Mitra, S., & Hayashi, Y. (2000). Neuro-fuzzy rule generation: survey in soft computing framework. *Neural Networks, IEEE Transactions on*, *11*(3), 748-768.

Mitra, S., & Kuncheva, L. (1995). Improving classification performance using fuzzy MLP and two-level selective partitioning of the feature space. *Fuzzy Sets and Systems*, *70*(1), 1-13. doi: http://dx.doi.org/10.1016/0165-0114(94)00244-2

Mucientes, M., Vidal, J., Bugarín, A., & Lama, M. (2009). Processing time estimations by variable structure TSK rules learned through genetic programming. *Soft Computing*, *13*(5), 497-509. doi: 10.1007/s00500-008-0364-2

Munoz-Salinas, R., Aguirre, E., Cordon, O., & Garcia-Silvente, M. (2008). Automatic Tuning of a Fuzzy Visual System Using Evolutionary Algorithms: Single-Objective Versus Multiobjective Approaches. *Trans. Fuz Sys.*, *16*(2), 485-501. doi: 10.1109/tfuzz.2006.889954

Nakaoka, K., Furuhashi, T., & Uchikawa, Y. (1994, 26-29 Jun 1994). *A study on apportionment of credits of fuzzy classifier system for knowledge acquisition of large scale systems.* Paper presented at the Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on.

Narukawa, K., Nojima, Y., & Ishibuchi, H. (2005). Modification of evolutionary multiobjective optimization algorithms for multiobjective design of fuzzy rule-based classification systems. *Fuzz-Ieee 2005: Proceedings of the Ieee International Conference on Fuzzy Systems*, 809-814.

Nauck, D. (2000). *Data Analysis with Neuro-Fuzzy Methods.* (Habilitation Thesis), University of Magdeburg.

Nauck, D. (2001, 25-28 July 2001). *Fuzzy data analysis with NEFCLASS.* Paper presented at the IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th.

Nauck, D. (2003, 25-28 May 2003). *Measuring interpretability in rule-based classification systems.* Paper presented at the Fuzzy Systems, 2003. FUZZ '03. The 12th IEEE International Conference on.

Nauck, D., Klawoon, F., & Kruse, R. (1997). *Foundations of Neuro-Fuzzy Systems*. New York John Wiley.

Nauck, D., & Kruse, R. (1995). *NEFCLASS- a neuro-fuzzy approach for the classification of data*. Paper presented at the Proceedings of the 1995 ACM symposium on Applied computing, Nashville, Tennessee, USA.

Nauck, D., & Kruse, R. (1997). A neuro-fuzzy method to learn fuzzy classification rules from data. *Fuzzy Sets and Systems, 89*(3), 277-288.

Nauck, D., & Kruse, R. (1999a). Neuro-fuzzy systems for function approximation. *Fuzzy Sets and Systems, 101*(2), 261-271.

Nauck, D., & Kruse, R. (1999b). Obtaining interpretable fuzzy classification rules from medical data. *Artificial Intelligence in Medicine, 16*(2), 149.

Nojima, Y., & Ishibuchi, H. (2006, Dec. 2006). *Designing Fuzzy Ensemble Classifiers by Evolutionary Multiobjective Optimization with an Entropy-Based Diversity Criterion.* Paper presented at the Hybrid Intelligent Systems, 2006. HIS '06. Sixth International Conference on.

Nojima, Y., & Ishibuchi, H. (2009). Incorporation of user preference into multi-objective genetic fuzzy rule selection for pattern classification problems. *Artificial Life and Robotics, 14*(3), 418-421. doi: 10.1007/s10015-009-0700-3

Nojima, Y., Mihara, S., & Ishibuchi, H. (2010). *Ensemble classifier design by parallel distributed implementation of genetic fuzzy rule selection for large data sets.* Paper presented at the Evolutionary Computation (CEC), 2010 IEEE Congress on.

Nojima, Y., Takahashi, Y., & Ishibuchi, H. (2014, 3-6 Dec. 2014). *Genetic lateral tuning of membership functions as post-processing for hybrid fuzzy genetics-based machine learning.* Paper presented at the Soft Computing and Intelligent Systems (SCIS), 2014 Joint 7th International Conference on and Advanced Intelligent Systems (ISIS), 15th International Symposium on.

Nozaki, K., Ishibuchi, H., & Tanaka, H. (1997). A simple but powerful heuristic method for generating fuzzy rules from numerical data. *Fuzzy Sets and Systems, 86*(3), 251-270. doi: Doi 10.1016/0165-0114(95)00413-0

Olaru, C., & Wehenkel, L. (2003). A complete fuzzy decision tree technique. *Fuzzy Sets and Systems, 138*(2), 221-254.

Paiva, R. P., & Dourado, A. (2004). Interpretability and learning in neuro-fuzzy systems. *Fuzzy Sets and Systems*, *147*(1), 17-38.

Pal, S. K., & Mitra, S. (1992). Multilayer perceptron, fuzzy sets, and classification. *IEEE Transactions on Neural Networks*, *3*(5), 683-697.

Park, D., Kandel, A., & Langholz, G. (1994). Genetic-Based New Fuzzy-Reasoning Models with Application to Fuzzy Control. *Ieee Transactions on Systems Man and Cybernetics*, *24*(1), 39-47. doi: Doi 10.1109/21.259684

Pedrycz, W., & Gomide, F. (1998). *An introduction to fuzzy sets: analysis and design*. Cambridge, MA. : MIT Press.

Pedrycz, W., & Sosnowski, Z. A. (2000). Designing decision trees with the use of fuzzy granulation. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, *30*(2), 151-159.

Pedrycz, W., & Sosnowski, Z. A. (2001). The design of decision trees in the framework of granular data and their application to software quality models. *Fuzzy Sets and Systems*, *123*(3), 271-290.

Pedrycz, W., & Sosnowski, Z. A. (2005). C-fuzzy decision trees. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, *35*(4), 498-511.

Peña-Reyes, C. A., & Sipper, M. (2003). Fuzzy CoCo: Balancing Accuracy and Interpretability of Fuzzy Models by Means of Coevolution. In J. Casillas, O. Cordón, F. Herrera & L. Magdalena (Eds.), *Interpretability Issues in Fuzzy Modelling*. Heidelberg, Germany: Springer-Verlag.

Piramuthu, S. (1999). Financial credit-risk evaluation with neural and neurofuzzy systems. *European Journal of Operational Research*, *112*(2), 310-321.

Pulkkinen, P., Hytonen, J., & Kolvisto, H. (2008). Developing a bioaerosol detector using hybrid genetic fuzzy systems. *Engineering Applications of Artificial Intelligence*, *21*(8), 1330-1346. doi: DOI 10.1016/j.engappai.2008.01.006

Pulkkinen, P., & Koivisto, H. (2008). Fuzzy classifier identification using decision tree and multiobjective evolutionary algorithms. *International Journal of Approximate Reasoning*, *48*(2), 526-543. doi: DOI 10.1016/j.ijar.2007.10.004

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, *1*(1), 81-106.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*: Morgan Kaufmann Publishers.

Riid, A. (2002). *Transparent fuzzy systems: modelling and control.* (PhD), Tallinn Technical University, Estonia.

Rodríguez, J. J., & Kuncheva, L. (2007). Naïve Bayes Ensembles with a Random Oracle. In M. Haindl, J. Kittler & F. Roli (Eds.), *Multiple Classifier Systems* (Vol. 4472, pp. 450-458): Springer Berlin Heidelberg.

Rokach, L. (2009). Taxonomy for characterizing ensemble methods in classification tasks: A review and annotated bibliography. *Computational Statistics & Data Analysis, 53*(12), 4046-4072. doi: http://dx.doi.org/10.1016/j.csda.2009.07.017

Rokach, L., Romano, R., & Maimon, O. (2008). Negation recognition in medical narrative reports. *Information Retrieval, 11*(6), 499-538.

Roubos, H., & Setnes, M. (2001). Compact and transparent fuzzy models and classifiers through iterative complexity reduction. *Fuzzy Systems, IEEE Transactions on, 9*(4), 516-524. doi: 10.1109/91.940965

Ruspini, E. H. (1969). A new approach to clustering. *Information and control, 15*(1), 22-32.

Sanchez, L., Couso, I., & Corrales, J. A. (2001). Combining GP operators with SA search to evolve fuzzy rule based classifiers. *Information Sciences, 136*(1-4), 175-191. doi: Doi 10.1016/S0020-0255(01)00146-3

Schapire, R. E., & Singer, Y. (1999). Improved Boosting Algorithms Using Confidence-rated Predictions. *Machine Learning, 37*(3), 297-336. doi: 10.1023/A:1007614523901

Setnes, M., Babuska, R., Kaymak, U., & Van Nauta Lemke, H. R. (1998). Similarity measures in fuzzy rule base simplification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 28*(3), 376-386. doi: 10.1109/3477.678632

Setnes, M., & Roubos, H. (2000). GA-fuzzy modeling and classification: complexity and performance. *Fuzzy Systems, IEEE Transactions on, 8*(5), 509-522. doi: 10.1109/91.873575

Setzkorn, C., & Paton, R. C. (2005). On the use of multi-objective evolutionary algorithms for the induction of fuzzy classification rule systems. *Biosystems, 81*(2), 101-112. doi: DOI 10.1016/j.biosystems.2005.02.003

Shann, J., & Fu, H. (1995). A fuzzy neural network for rule acquiring on fuzzy control systems. *Fuzzy Sets and Systems*, *71*(3), 345-357.

Shi, Y., Eberhart, R., & Chen, Y. (1999). Implementation of evolutionary fuzzy systems. *Fuzzy Systems, IEEE Transactions on*, *7*(2), 109-119. doi: 10.1109/91.755393

Shimojima, K., Fukuda, T., & Hasegawa, Y. (1995). Self-Tuning Fuzzy Modeling with Adaptive Membership Function, Rules, and Hierarchical Structure-Based on Genetic Algorithm. *Fuzzy Sets and Systems*, *71*(3), 295-309. doi: Doi 10.1016/0165-0114(94)00280-K

Srinivas, N., & Deb, K. (1994). Muiltiobjective optimization using nondominated sorting in genetic algorithms. *Evol. Comput.*, *2*(3), 221-248. doi: 10.1162/evco.1994.2.3.221

Street, W. N., Wolberg, W. H., & Mangasarian, O. L. (1993). *Nuclear feature extraction for breast tumor diagnosis.* Paper presented at the IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology.

Sugeno, M., & Kang, G. T. (1988). Structure identification of fuzzy model. *Fuzzy Sets and Systems, 28*(1), 15-33. doi: http://dx.doi.org/10.1016/0165-0114(88)90113-3

Sulzberger, S. M., Tschichold-Gurman, N., & Vestli, S. J. (1993). *FUN: Optimization of fuzzy rule based systems using neural networks.* Paper presented at the Neural Networks, 1993., IEEE International Conference on.

Sung, T. K., Chang, N., & Lee, G. (1999). Dynamics of Modeling in Data Mining: Interpretive Approach to Bankruptcy Prediction. *Journal of Management Information Systems*, *16*(1), 63-85. doi: 10.2307/40398418

Takagi, H., Suzuki, N., Koda, T., & Kojima, Y. (1992). Neural networks designed on approximate reasoning architecture and their applications. *Neural Networks, IEEE Transactions on*, *3*(5), 752-760.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *Systems, Man and Cybernetics, IEEE Transactions on*(1), 116-132.

Tano, S., Oyama, T., & Arnould, T. (1996). Deep combination of fuzzy inference and neural network in fuzzy inference software — FINEST. *Fuzzy Sets and Systems*, *82*(2), 151-160. doi: http://dx.doi.org/10.1016/0165-0114(95)00251-0

Thrift, P. (1991). *Fuzzy logic synthesis with genetic algorithms.* Paper presented at the Proceedings of the fourth international conference on genetic algorithms.

Tikk, D., Gedeon, T. D., & Wong, K. W. (2003). A feature ranking algorithm for fuzzy modelling problems. In J. Casillas, O. Cordón, F. Herrera & L. Magdalena (Eds.), *Interpretability Issues in Fuzzy Modelling* (pp. 176-192). Heidelberg, Germany: Springer-Verlag.

Trawinski, K., Cordon, O., & Quirin, A. (2011). On designing fuzzy rule-based multiclassification systems by combining FURIA with bagging and feature selection. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 19*(04), 589-633. doi: doi:10.1142/S0218488511007155

Trawinski, K., Cordon, O., & Quirin, A. (2013, 7-10 July 2013). *Random oracles fuzzy rule-based multiclassifiers for high complexity datasets.* Paper presented at the Fuzzy Systems (FUZZ), 2013 IEEE International Conference on.

Trawinski, K., Cordon, O., & Quirin, A. (2014, 6-11 July 2014). *Embedding evolutionary multiobjective optimization into fuzzy linguistic combination method for fuzzy rule-based classifier ensembles.* Paper presented at the Fuzzy Systems (FUZZ-IEEE), 2014 IEEE International Conference on.

Trawiński, K., Cordón, O., Quirin, A., & Sánchez, L. (2013a). Multiobjective genetic classifier selection for random oracles fuzzy rule-based classifier ensembles: How beneficial is the additional diversity? *Knowledge-Based Systems*(0). doi: http://dx.doi.org/10.1016/j.knosys.2013.08.006

Trawiński, K., Cordón, O., Quirin, A., & Sánchez, L. (2013b). Multiobjective genetic classifier selection for random oracles fuzzy rule-based classifier ensembles: How beneficial is the additional diversity? *Knowledge-Based Systems, 54*(0), 3-21. doi: http://dx.doi.org/10.1016/j.knosys.2013.08.006

Trawinski, K., Cordon, O., Sanchez, L., & Quirin, A. (2013). A Genetic Fuzzy Linguistic Combination Method for Fuzzy Rule-Based Multiclassifiers. *Fuzzy Systems, IEEE Transactions on, 21*(5), 950-965. doi: 10.1109/TFUZZ.2012.2236844

Trawinski, K., Quirin, A., & Cordon, O. (2009). Bi-criteria Genetic Selection of Bagging Fuzzy Rule-based Multiclassification Systems. *Proceedings of the Joint 2009 International Fuzzy Systems Association World Congress and 2009 European Society of Fuzzy Logic and Technology Conference,* 1514-1519.

Trawinski., Cordón, O., & Quirin, A. (2012). A Study on the Use of Multiobjective Genetic Algorithms for Classifier Selection in FURIA-based Fuzzy

Multiclassifiers. *International Journal of Computational Intelligence Systems, 5*(2), 231-253. doi: 10.1080/18756891.2012.685272

Triviño-Rodriguez, J. L., Ruiz-Sepúlveda, A., & Morales-Bueno, R. (2008). How an ensemble method can compute a comprehensible model *Data Warehousing and Knowledge Discovery* (pp. 368-378): Springer.

Tsakonas, A. (2006). A comparison of classification accuracy of four genetic programming-evolved intelligent structures. *Information Sciences, 176*(6), 691-724. doi: DOI 10.1016/j.ins.2005.03.012

Van Broekhoven, E., Adriaenssens, V., & De Baets, B. (2007). Interpretability-preserving genetic optimization of linguistic terms in fuzzy models for fuzzy ordered classification: An ecological case study. *International Journal of Approximate Reasoning, 44*(1), 65-90.

Vanhoucke, V., & Silipo, R. (2003). Interpretability in multidimensional classification. In J. Casillas, O. Cordón, F. Herrera & L. Magdalena (Eds.), *Interpretability Issues in Fuzzy Modelling* (Vol. 128, pp. 193-220). Heidelberg, Germany Springer-Verlag.

Vélez, M. Á., Sánchez, O., Romero, S., & Andújar, J. M. (2010). A new methodology to improve interpretability in neuro-fuzzy TSK models. *Applied Soft Computing, 10*(2), 578-591.

Venturini, G. (1993). SIA: A supervised inductive algorithm with genetic search for learning attributes based concepts. In P. Brazdil (Ed.), *Machine Learning: ECML-93* (Vol. 667, pp. 280-296). Heidelberg: Springer-Verlag

W., H., F., W., Y., P. S., & H., J. (2003). *Mining concept-drifting data streams using ensemble classifiers*. Paper presented at the Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, Washington, D.C.

Wall, R., Cunningham, P., Walsh, P., & Byrne, S. (2003). Explaining the output of ensembles in medical decision support on a case by case basis. *Artificial Intelligence in Medicine, 28*(2), 191-206. doi: http://dx.doi.org/10.1016/S0933-3657(03)00056-3

Wang, H., Kwong, S., Jin, Y., Wei, W., & Man, K. F. (2005a). Agent-based evolutionary approach for interpretable rule-based knowledge extraction. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 35*(2), 143-155. doi: 10.1109/TSMCC.2004.841910

Wang, H., Kwong, S., Jin, Y., Wei, W., & Man, K. F. (2005b). Multi-objective hierarchical genetic algorithm for interpretable fuzzy rule-based knowledge extraction. *Fuzzy Sets and Systems, 149*(1), 149-186. doi: http://dx.doi.org/10.1016/j.fss.2004.07.013

Wang, L., & Yen, J. (1999). Extracting fuzzy rules for system modeling using a hybrid of genetic algorithms and Kalman filter. *Fuzzy Sets Syst., 101*(3), 353-362. doi: 10.1016/s0165-0114(97)00098-5

Wang, L. X. (2003). The WM method completed: a flexible fuzzy system approach to data mining. *Fuzzy Systems, IEEE Transactions on, 11*(6), 768-782. doi: 10.1109/TFUZZ.2003.819839

Wang, L. X., & Mendel, J. M. (1992). Generating fuzzy rules by learning from examples. *Systems, Man and Cybernetics, IEEE Transactions on, 22*(6), 1414-1427. doi: 10.1109/21.199466

Wang, X., & Wang, H. (2006). Classification by evolutionary ensembles. *Pattern Recognition, 39*(4), 595-607. doi: http://dx.doi.org/10.1016/j.patcog.2005.09.016

Weber, R. (1992). *Fuzzy-ID3: a class of methods for automatic knowledge acquisition.* Paper presented at the The second international conference on fuzzy logic and neural networks.

West, D. (2000). Neural network credit scoring models. *Computers & Operations, 27*, 1131-1152.

West, D., Mangiameli, P., Rampal, R., & West, V. (2005). Ensemble strategies for a medical diagnostic decision support system: A breast cancer diagnosis application. *European Journal of Operational Research, 162*(2), 532-551. doi: http://dx.doi.org/10.1016/j.ejor.2003.10.013

Whitley, D. (1994). A genetic algorithm tutorial. *Statistics and Computing, 4*(2), 65-85. doi: 10.1007/BF00175354

Woods, K. S., Solka, J. L., Priebe, C. E., Doss, C. C., Bowyer, K. W., & Clarke, L. P. (1993). Comparative evaluation of pattern recognition techniques for detection of microcalcifications. 841-852. doi: 10.1117/12.148696

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., . . . Steinberg, D. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems, 14*(1), 1-37. doi: 10.1007/s10115-007-0114-2

Xu, L., Krzyzak, A., & Suen, C. Y. (1992). Methods of combining multiple classifiers and their applications to handwriting recognition. *Systems, Man and Cybernetics, IEEE Transactions on, 22*(3), 418-435. doi: 10.1109/21.155943

Yager, R. R., & Filev, D. P. (1994). *Essentials of fuzzy modeling and control*. New York, NY: John Wiley.

Yang, H., & Moody, J. (1999). Data visualization and feature selection: New algorithms for non-gaussian data. *in Advances in Neural Information Processing Systems* (Vol. 12, pp. 687--693): MIT Press.

Yaochu, J. (2000). Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement. *Fuzzy Systems, IEEE Transactions on, 8*(2), 212-221. doi: 10.1109/91.842154

Yen, J., & Wang, L. (1999). Simplifying fuzzy rule-based models using orthogonal transformation methods. *Trans. Sys. Man Cyber. Part B, 29*(1), 13-24. doi: 10.1109/3477.740162

Zadeh, L. A. (1965). Fuzzy sets. *Information and control, 8*(3), 338-353.

Zhou, S. M., & Gan, J. Q. (2008). Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modelling. *Fuzzy Sets and Systems, 159*(23), 3091-3131. doi: http://dx.doi.org/10.1016/j.fss.2008.05.016

Zitzler, E., Laumanns, M., & Thiele, L. (2001). *SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization.* Paper presented at the International Center for Numerical Methods in Engineering (CIMNE).

Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the Strength Pareto approach. *IEEE Transactions on Evolutionary Computation, 3*(4), 257-271. doi: Doi 10.1109/4235.797969

# LIST OF PUBLICATIONS

## Articles

Lahsasna, A., Ainon, R. N., Zainuddin, R., & Bulgiba, A. (2012). Design of a fuzzy-based decision support system for coronary heart disease diagnosis.*Journal of medical systems*, *36*(5), 3293-3306.

Ainon, R. N., Bulgiba, A. M., & Lahsasna, A. (2012). AMI screening using linguistic fuzzy rules. *Journal of medical systems*, *36*(2), 463-473.

Lahsasna, A., Ainon, R. N., & Teh, Y. W. (2010). Credit Scoring Models Using Soft Computing Methods: A Survey. *Int. Arab J. Inf. Technol.*, *7*(2), 115-123.

Lahsasna, A., Ainon, R. N., & Wah, T. Y. (2010). Enhancement of transparency and accuracy of credit scoring models through genetic fuzzy classifier. *Maejo International Journal of Science and Technology*, *4*(1), 136-158.

## Book Chapter

Lahsasna, A., Ainon, R. N., Zainuddin, R., & Bulgiba, A. M. (2012). A transparent fuzzy rule-based clinical decision support system for heart disease diagnosis. In *Knowledge Technology* (pp. 62-71). Springer Berlin Heidelberg.

## Conference Proceedings

Adel Lahsasna, Raja Noor Ainon, Roziati Zainuddin Awang M. Bulgiba, "A Transparent Fuzzy rule-based Clinical Decision Support System for Heart Disease Diagnosis", Third Malaysian Joint Conference on Artificial Intelligence (MJCAI2011), 20-22 July 2011.

Saybani, M. R., Wah, T. Y., Lahsasna, A., Amini, A., & Aghabozorgi, S. R. (2011, November). Data mining techniques for predicting values of a faulty sensor at a refinery. In *Computer Sciences and Convergence Information Technology (ICCIT), 2011 6th International Conference on* (pp. 792-796). IEEE.

Ainon, R. N., Lahsasna, A., & Wah, T. Y. (2009, May). A transparent classification model using a hybrid soft computing method. In *Modelling & Simulation, 2009. AMS'09. Third Asia International Conference on* (pp. 146-151). IEEE.

Lahsasna, A., Ainon, R. N., & Wah, T. Y. (2008, May). Intelligent credit scoring model using soft computing approach. In *Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on*(pp. 396-402). IEEE.

Lahsasna, A., Ainon, R. N., & Wah, T. Y. (2008, August). Credit risk evaluation decision modeling through optimized fuzzy classifier. In*Information Technology, 2008. ITSim 2008. International Symposium on*(Vol. 1, pp. 1-8). IEEE.