Amaury Teófilo Brasil Filho

A Novel Approach Based on Multiple Criteria Decision Aiding Methods to Cope with Classification Problems

Fortaleza, CE
2009

Amaury Teófilo Brasil Filho

A Novel Approach Based on Multiple Criteria Decision Aiding Methods to Cope with Classification Problems

Dissertation presented to the board of faculties of the Graduate Program in Applied Informatics at the University of Fortaleza, as partial fulfillment of the requirements for the Master's degree in Applied Informatics.

Supervisor: Plácido Rogério Pinheiro
Co-Supervisor: André Luís Vasconcelos Coelho

Fortaleza, CE
2009

Abstract

This work presents a novel decision making approach developed upon two Multicriteria Decision Aiding (MCDA) classification methods. The classification methods are centered on the concept of prototypes, that are alternatives that serve as class representatives related to a given problem, and have their performance index very dependent upon the choice of values of some control parameters. In contrast with most MCDA classification research, which put more emphasis on the development of new methods, this study investigates the impact that the prototype selection task exerts on the classification performance exhibited by the classifiers under analysis. To overcome this problem and leverage the classifier performance, two techniques, one based on ELECTRE IV methodology and the other on a customized genetic algorithm, are employed in order to select the prototypes and calibrate the control parameters. To validate the devised classification approach, a new dataset has been designed based on the global consensus that concerns the AD diagnosis. In this context, an Alzheimer's disease (AD) case study was developed to assist clinicians and researchers in the AD early detection. Moreover, the experiments realized, involving different datasets, reveal that there are still some gaps in existing MCDA classification methods that can lead to an improvement of the methodology for the classification problems.

Keywords: Multicriteria Decision Aiding, Classification, ELECTRE IV, Genetic Algorithm, Alzheimer's Disease.

Resumo

Esse trabalho apresenta uma nova abordagem de tomada de decisão desenvolvida sobre dois métodos Multicritério de Apoio à Decisão (MCDA) na área de classificação. Os métodos de classificação são fundamentados no conceito de protótipos, que são alternativas que funcionam como representantes de classe de um determinado problema, e possuem sua performance dependente da escolha dos parâmetros de controle do algoritmo. Em contraste com a maioria das pesquisas relacionadas à MCDA, que colocam uma maior ênfase no desenvolvimento de novos métodos, esse estudo investiga o impacto que a tarefa de seleção de protótipos exerce na performance dos algoritmos de classificação em questão. Para superar esse problema e melhorar a performance dos classificadores, duas técnicas, uma baseada na metodologia ELECTRE IV e a outra em um algoritmo genético customizado, foram aplicadas para selecionar os protótipos e calibrar os parâmetros de controle. Para validar a abordagem de classificação criada, um novo dataset foi elaborado tendo como base os consensos mundiais relacionados ao diagnóstico de Alzheimer. Nesse contexto, um estudo de caso aplicado à doença de Alzheimer foi desenvolvido para ajudar médicos e pesquisadores na identificação precoce da doença. Além disso, os experimentos realizados, envolvendo diferentes conjuntos de dados, revelam que existem alguns espaços para melhoria nos métodos de classificação MCDA que podem levar a um avanço na metodologia dos problemas de classificação.

Palavras-chave: Decisão Multicritério, Classificação, ELECTRE IV, Algoritmo Genético, Doença de Alzheimer.

Acknowledgments

I would like to give my sincere thanks to all those who helped me to carry out this research.

I wish to express my deep gratitude to Professor Plácido Pinheiro, who introduced me to the undergraduate research. After this period, he suggested me to expand my research beginnings to the postgraduate studies. During this work he provided a continuous assistance and encouragement, constructive guidance, kind patience, and moral support, that could make me able to achieve my objective. I am especially grateful to Professor André Coelho, for all the considerations that came to enhance this work. Without his constructive comments, guidance, questions, and advices this work would not be the same. It is my great honor to work under their supervision and have my self-confidence strengthened through this experience.

I would like to express my appreciations to the professionals of the Nucleus of Application in Information Technology of the University of Fortaleza for the exchange of knowledge during this work, as well as to Ricardo Lima and Nathanael Costa for their support and involvement.

I wish to acknowledge FUNCAP for the financial support during my postgraduate research.

I am also thankful to my friends for bringing joy and spirit into my life and sharing wonderful moments with me.

I cannot end without thanking my family, my sisters, Mariana and Mônica, my fiancee, Juliana, and especially, my dear parents, Amaury and Eliana for their constant encouragement and love, on which I have relied throughout these years.

To my parents, Amaury and Eliana, and my fiancee, Juliana, the love of my life.

SUMMARY

List of Figures

List of Tables

Glossary

AD - Alzheimer's Disease

CERAD - Consortium to Establish a Registry for Alzheimer's Disease

CI - Concordance Index

DM - Decision Maker

ELECTRE - Elimination Et Choix Traduisant la Realité

FPV - Fundamental Point of View

GA - Genetic Algorithm

gMCDA - Goletsis' Multicriteria Decision Analysis

IBL - Instance Based Learning

MAUT - Multiattribute Utility Theory

MCDA - Multicriteria Decision Aiding

MCDM - Multicriteria Decision Making

MD - Membership Degree

MMSE - Mini-Mental State Examination

NCM - Nominal Classification Methods

ORCLASS - Ordinal Classification

ORT - Outranking Utility Theory

PACOM - Paired Compensation

PDA - Preference Disaggregation Approach

SAC - Subset Alternative Classification

SI - Similarity Index

SPECT - Single Photon Emission Computed Tomography

UT - Utility Function

UTA - Utilités Additives

UTADIS - Utilités Additives Discriminantes

VDA - Verbal Decision Analysis

List of Works Published by the Author

1. A.T. Brasil, P.R. Pinheiro, and A.L.V. Coelho. *Innovations and Advances in Computer Sciences and Engineering*, chapter "The Impact of the Prototype Selection on a Multicriteria Decision Aid Classification Algorithm" (to appear), Springer Verlag, 2009.

2. A.T. Brasil Filho, P.R. Pinheiro, and A.L.V. Coelho. "Towards the Early Diagnosis of Alzheimer's Disease via a Multicriteria Classification Model". *5th Evolutionary Multi-Criterion Optimization* (EMO 2009). Lecture Notes in Computer Science, Vol. 5467, pg. 393-406, Nantes, France, April 2009.

3. A.T. Brasil Filho, P.R. Pinheiro, A.L.V. Coelho, and N.C. Costa. "Comparison of Two MCDA Classification Methods over the Diagnosis of Alzheimer's Disease". *Fourth International Conference on Rough Set and Knowledge Technology* (RSKT 2009). Lecture Notes in Computer Science, Vol. 5589, pg. 289-298, Gold Cost, Australia, 2009.

4. A.T. Brasil Filho, P.R. Pinheiro, A.L.V. Coelho, and N.C. Costa. "Comparison of Two Prototype-Based Multicriteria Classification Methods". *IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making* (MCDM 2009), Nashville, USA, 2009.

5. A.T. Brasil Filho, P.R. Pinheiro, A.L.V. Coelho, and N.C. Costa. "Selecting Prototypes for Two Multicriteria Classification Methods: A Comparative Study". *IEEE World Congress on Nature and Biologically Inspired Computing* (NABIC 2009), Coimbatore, India, 2009.

Introduction

The decision making task is associated with people's life in their daily activities. However, besides the fact of performing this activity continuously, many persons still present some difficulties when dealing with it. Normally, as a result of a bad decision making, these people demonstrate some kind of repentance. The problem even gets worse when the decision making encloses money, business world, and health. For such cases, a bad decision can bring not just a momentary repentance, but irreparable losses.

Trying to tackle this type of problem, many researchers have dedicated their efforts to assist these people during the decision aiding process. This process, according to the nature of the problem, the policy of the decision maker, and the overall objective of the decision, can be grouped into four distinct approaches: ranking, choice, description, and classification [77]. For example, sometimes the decisor may require the choice of an alternative among a group of alternatives, or maybe he or she wants to rank a set of alternatives according to his/her preferences.

The focus of this study is directed to the classification problem, which consists in assigning an alternative to a set of categories or classes. This problematic has been one of the most investigated during the past decades. Some authors [77, 122] dedicate this relevance to the classification problems due to their practical interest. Therefore, many works contemplated the development of methods to solve this type of problem in an effective way.

At the beginning, multivariate statistical analysis techniques such as discriminant analysis (linear and quadratic), and econometric techniques such as logit and probit analysis, the linear probability model, etc., have dominated this field [77]. As some questionings, enclosing the applicability of these methods, started to appear, new classification approaches were developed over two main disciplines: artificial intelligence and operations research. Rapidly, methods based on neural networks, machine learning, fuzzy sets as well as multicriteria decision aiding (MCDA) introduced new features that could bring superior results.

Among these techniques, this work has its research based on the MCDA theory. The term multicriteria refers to the possibility of estimating an alternative under different perspectives (points of view), and evaluate it based on the decisor's preferences. These are the main features that provide a decision support orientation to the decisor, distinguishing the multicriteria classification theory from other approaches.

Even though the research enclosing the multicriteria classification has evolved over the last decades, there are still some unsolved problems left and interesting research topics that require further and deeper investigation [55]. Indeed, as pointed out by Zopounidis and Doumpos [23], the great majority of works conducted on the MCDA classification theme had their focus on the development of novel MCDA classification methods, not giving much emphasis on characterizing and comparing their distinctive aspects.

In this context, the objective of this dissertation is to develop an approach to assist the decision maker and leverage the classification rates during the employment of multicriteria nominal classifiers. In such regard, two methods from different research areas that can lead to an improvement of the classifiers' performance were employed. Besides the various experiments performed over some well-known datasets, a case study to assist in the early detection of Alzheimer's disease and to validate the overall methodology was developed.

The rest of the work is organized as follows. In Chapter 1, the main concepts and definitions enclosing the multicriteria domain are introduced. Moreover, a literature review of the MCDA theory to provide the necessary foundation and support for the research is performed. Chapter 2 details some of the most important MCDA classification methods developed under the MCDA context. Chapter 3 presents some experiments developed with two nominal classification methods that evidenced some gaps in existing MCDA methods. Moreover, this chapter describes the proposed approach, its main methodologies, and the achieved results. In Chapter 4, the Alzheimer's disease is introduced and the case study developed upon this pathology is detailed. In addition, the experiments and the achieved results are discussed. Finally, some conclusions and directions for future research are drawn.

Chapter 1

Multicriteria Decision Aiding Preliminaries

The purpose of this chapter is to present the basic concepts and a literature review of the multicriteria decision aiding theory to provide the necessary foundation for the research in this work. The chapter begins by defining the concepts that play a fundamental role for analyzing and structuring decision problems. In this context, different preference relations, problem types, and other relevant research topics are briefly explained. At the end of the chapter the objectives and methodologies related to this study are informed.

1.1 Decision Aid

Decision aiding is a daily activity of almost everyone, and the implications of a decision making can result in many unexpected consequences. Some people present difficulties even in simple everyday acts, as the choice of a meal in a restaurant or a social activity, for example.

The study of decision making is part of many disciplines, including psychology, business, engineering, operations research, systems engineering, and management science [122]. As society becomes more complex, the need for decisions that balance conflicting objectives (criteria) has grown. For example, when the decision is related to the business world the problem increases in complexity and faces strategic decisions that can bring undesirable results. Roy [15] defined decision aiding as follows: decision aiding is the activity of the person who, through the use of explicit, but not necessarily completely formalized models, helps obtain elements of responses to the questions posed by a stakeholder in a decision process.

Decision is related to the act of making a choice among different alternatives that present conflicting points of view. The decision aiding helps on this choice by making estimatives and comparing those conflicting criteria. When a problem is composed by more than one path to its solution, we can say that we are facing a decision making situation. Even in situations where there is only one action to be taken, many different alternatives can be found to take it or not. The pros and the cons of distinct points of view is the domain of MCDA [36]. The multicriteria concept refers to the vast variety of tools and methodologies developed with the intent of helping a decision maker (DM) to select from finite sets of alternatives according to two or more criteria, which are usually conflicting.

In multicriteria there are a couple of designations to describe these types of decision assistance such as: "multiple criteria decision aiding", which was originated in the European multicriteria school [15], and "multiple objective decision making", which is widely used in North America.

In opinion of Gomes et al. [74], MCDA does not aim to find a solution for the problem, electing only the truth that is represented by the selected alternative. They advocate that, in a general way, inside of a multicriteria problem there is not an alternative or solution that is better considering all criteria simultaneously.

In this context, the rest of this chapter will be dedicated to the fundamental concepts that are related to the MCDA theory.

1.2 The Structure of Multicriteria Decision Aiding Problems

1.2.1 Actors

In multicriteria, the actors are the people that take part of the problem, acting in different moments and activities to assist in the decision process. Commonly, at least two persons are involved in the process of decision aiding: the DM and the analyst. For Mello et al. [65] it is important to observe that the definitions concerning the roles of the stakeholders are merely didactic.

The DM is a person or a group of persons who is being supported by the MCDA methodologies and who takes the responsibility for ratifying the final decision and for assuming its consequences. The DM is not supposed to know about MCDA tools or methods, however, his responsibility for the final decision forces him to understand the general structure of the problem and establish its limits [32, 74].

According to Yevseyeva [55], usually, the DM should be able to define the main elements required for the model, to express the ideas and the objectives to be reached, knowledge and preferences. In some situations, instead of a single DM, there is a group of persons taking part in the decision process. In such cases, it is necessary to distinguish the degree of influence of each DM considering the aggregation of preferences of the entire group.

Another person that exerts a determinant position in the MCDA context is the analyst. The definitions of the analyst's main activities can be described as follows [55]:

- Interprets and quantifies the DM's opinions;
- Structures the problem;
- Helps the DM to acquire expert's knowledge;
- Elaborates the MCDA problem model;
- Chooses a method to find the solution that best matches the DM's objectives;
- Presents the results for the decision.

During the decision process the analyst should act in constant interaction with the decisors, trying to model their needs to provide the best results. The analyst is usually a specialist over a certain domain who is familiar with MCDA techniques, understanding their definitions and capabilities in order to provide a well-defined model.

Usually, the decision aiding processes regard the experts as other stakeholders. Differently from the analyst, that have his or her focus on the MCDA model, they have their professional expertise in the problem area or some part of it. This way, they influence the decision by providing their knowledge or by helping in the model's construction. Besides the fact that the expert does not provide the final solution, he or she can provide all the necessary knowledge for the DM to support his or her final decision [55]. However, in this work, the DM and the expert are regarded as the same person (and called the DM) .

1.2.2 Alternatives

According to Roy [15], a potential alternative is a realistic or fictitious alternative. The alternative is considered real or realistic if it exists in real life and fictious if it can assist in the decision aiding process and does not exist in real life, but is temporarily judged as being realistic by at least one actor. An action is qualified as potential when it is deemed possible to implement it, or when it deserves some interest within the decision aiding process. The potential alternative can be identified in the beginning or during the decision process, which can turn into a solution of the problem being studied [74].

A set of given alternatives, denoted as A, is the set of potential alternatives, actions or objects each of which should be estimated directly (by the DM) or indirectly (by some method) for the solution of the problem considered. Here, the terms alternatives, actions and objects are treated as synonyms. This set is not necessarily stable, it can vary throughout the decision process. Such an evolution may come from a dynamic environment, but also from the problem itself [16].

An alternative $a_i \in A$ is a vector of criteria values (or a tuple of attributes values). Usually, the set of alternatives is available or defined by the DM. As an extension of the categories established in [55], the set of alternatives can be divided in six categories:

- Finite: if it is possible to define all members of the set of alternatives;
- Infinite: if it is not possible to define all members of the set of alternatives;
- Stable: if the number and content of alternatives from the set of given alternatives cannot be changed during the decision aid process;
- Dynamic: if the set of alternatives evolves or changes during the decision aid process;
- Comprehensive: if accepting one alternative as a final decision excludes the possibility of accepting any other one;
- Fragmental: if a combination of alternatives can be the final decision.

1.2.3 Criteria and Attributes

In MCDA, the alternatives are estimated based on a set of criteria and/or attributes denoted as G. During the decision process, each criterion composes the features of each alternative being used to evaluate and compare potential actions according to a fundamental point of view (FPV). Each criterion or attribute $g_j \in G$ defines one feature of the alternative.

Criteria may have verbal or numerical values on their scales. Roy in [15] defines a criterion with numerical values on the scale as a mapping g_j from a set of alternatives A to a numerical scale (g_1,g_2,...,g_n), such that it appears meaningful to compare two alternatives a_i and a_r according to a particular FPV $g(a)$. The representation of a FPV (aspects, factors, features) constitutes one of the most difficult parts in a decision problem formulation. This implies that a criterion induces a preference relation on the set A.

Roy established the following set of properties which, if satisfied, defines the criteria as coherent [15].

- Completeness: if all criteria that allow distinction of alternatives from the set of given alternatives were taken into account. That means that there is no pair of alternatives, for which it is possible to say: "a_i is preferred to a_r , and a_r is preferred to a_i ", and for which the following preference relations are true: a_iPa_r and a_rPa_i. If this property is not satisfied, some important criteria are not taken into account, and we have an incomplete set of criteria;

- Cohesiveness: if two alternatives are indifferent (have the same values on all criteria), then improving value of one criterion of one alternative a_i and deteriorating value on some other criterion of some other alternative a_r would reflect the following DM's preferences "a_i is preferred to a_r" a_iPa_r;

- Non-redundancy: if each criterion from the set plays a significant role and removing at least one criterion leads to a violation of one of the two properties considered above.

When dealing with the classification approaches, the traditional methods (e.g., statistics and artificial intelligence) estimate the alternatives with attributes that are neither maximized nor minimized. According to Roy [15], the attribute defines "a simple characteristic or sign that serves as a basis of an estimation or assessment whose only purpose is to discern or distinguish", when compared to a criterion that serves as "a basis for preferential judgment". The attributes may have numerical or verbal values on their scales as well as have ordered values that, however, cannot be used for the preferential judgment but only distinguishing judgment [55].

For some types of classification problems, description of alternatives by attributes is much more natural. In an ideal case, the method should be able to consider both criteria and attributes(for instance, in the way that is done in the rough set approach) [55].

1.2.4 Relations

The preference modeling is one of the most important stages during the structuring of a decision problem [15, 32]. To provide substantial support for the decisors, they need to establish certain conditions to manifest a preference relation between the alternatives. These conditions are known by binary relations.

Makarov et al. [57] describe that a binary relation R over a set Ω is a subset $\Omega \times \Omega$, i.e., $R \subseteq \Omega \times \Omega$, with notation xRy for $(x, y \in \Omega)$. In what follows the classical properties of a binary relationship R over a set Ω are presented:

- It is reflexive if xRx is true;

- It is irreflexive if xRy implies $x \neq y$;
- It is symmetrical if xRy is true and yRx is true;
- It is asymmetric if at least one of the sentences, xRy and yRx, is false;
- It is transitive if xRz and zRy are true, and, xRy is also true.

In order to model the DM's preferences, it is necessary to define relations between the pairs of alternatives. At this moment the decisor can choose one out of the four basic binary relations: Strict Preference (P), Weak Preference (Q), Indifference (I), and Incomparability (R). These relations form the basic system of preference relations [15].

The indifference relation between two alternatives a_i and a_r, denoted as a_iIa_r, corresponds to the existence of clear and positive reasons that justify equivalence between these two alternatives [15]. This occurs when two alternatives a_i and a_r present the same importance to the DM. The indifference relation is reflexive and symmetric.

The strict preference relation of one alternative a_i over the other one a_r, denoted as a_iPa_r, corresponds to the existence of clear and positive reasons that justify significant preference in favor of one (well identified) of the two alternatives [15]. Such a situation appears when, for the DM, one alternative a_i is superior than the other a_r. The strict preference relation is asymmetric and non reflexive.

The weak preference relation, denoted as a_iQa_r, corresponds to the existence of clear and positive reasons that invalidate strict preference in favor of one (well identified) of the two alternatives but that are insufficient to deduce either strict preference in favor of the other alternative or indifference between two alternatives, thereby not allowing either of the two preceding situations to be distinguished as appropriate [15]. This situation appears when the DM is not sure yet to make a precise judgment about preference or indifference between two alternatives a_i and a_r. The weak preference relation is asymmetric and non reflexive.

The incomparability relation, denoted as a_iRa_r, corresponds to an absence of clear and positive reasons that justify any of the preceding relations [15]. In this situation, the alternative a_i is not in any of the above-mentioned relations with the alternative a_r, or the alternative a_r is not in any of the above-mentioned relations with the alternative a_i. The incomparability relation is symmetric and non reflexive.

Normally, in order to model the DM's preferences, the basic structure defined by Roy [15] is extended. This extension is called system of preference relations, that is composed by the following relations: non-preference (∼), preference (Pr), J-preference (J), K-preference (K), outranking (S) [15]. As described in Roy [15] and Gomes et al. [74] these relations can be detailed in the following way:

- Non-Preference (∼): Corresponds to an absence of clear and positive reasons to justify the strict or weak preference in favor of one alternative, and therefore, consolidates the situations of indifference or incomparability without being capable to differentiate them. $a_i \sim a_r \Longleftrightarrow a_iIa_r$ or a_iRa_r;
- Preference (Pr): Corresponds to the existence of clear and positive reasons that justify the strict or weak preference in favor of one (well identified) alternative, and therefore, consolidates the

strict and weak preference situations without being capable to differentiate them. $a_i \ Pr \ a_r \iff a_iPa_r$ or a_iQa_r;

- J-Preference (J): Corresponds to the existence of clear and positive reasons that justify a weak preference, independently of how weak it is, in favor of one (well identified) alternative, or in the limit of indifference between them, but in a way that no significant division be established between the situations of weak preference and indifference. $a_iJa_r \iff a_iQa_r$ or a_iIa_r;

- K-Preference (K): Corresponds to the existence of clear and positive reasons that justify the strict preference in favor of one (well identified) alternative or incomparability between them, but in a way that no significant division be established among the strict preference and incomparability situations. $a_iKa_r \iff a_iPa_r$ or a_iRa_r;

- Outranking: Corresponds to the existence of clear and positive reasons that justify the preference or J-preference in favor of one (well identified) alternative, but in a way that no significant division be established among the strict preference, weak preference and indifference situations. $a_iSa_r \iff a_iPa_r$ or a_iQa_r or a_iIa_r.

More information on other preference relations and their properties can be found in [15, 97].

When dealing with concepts enclosed in correlated multidisciplinary areas, such as artificial intelligence and multicriteria decision making (MCDM), their meaning and understanding can vary according to its domain. The work of Yevseyeva [55] emphasizes that from a classical point of view (e.g., artificial intelligence), in particular from expert systems, the concept of knowledge can be understood as all information acquired from an expert for constructing decision rules. Thereby, when adapted to MCDM, knowledge can be referred to as all kinds of information (e.g., initial information, or information obtained during the decision aiding process) that does not require the DM to make pairwise comparisons of elements (e.g., alternatives, criteria, criteria values, and classes) and to say which of the two elements is better. This sort of preferential information, obtained by the MCDA relations, is normally referred to as preferences.

In this context, this study will not make any sort of distinction over the information provided by the DM and/or the expert considering the meanings of knowledge and preferences. In addition, it refers to both types of information as preferences and do not treat them in any specific way. Next, the different types of MCDA problems that the DM may meet are considered.

1.2.5 MCDA Problematics

During the structuring phase of the decision aiding process it is important to obtain a right perception of how the stakeholders want to structure the problem. This activity will guide the DM through the most important step for studying a multiple criteria decision problem: the type of problem identification.

In MCDA theory, Roy [16] developed four goals in order to provide meaningful support to DMs. The first goal is related to the identification of the best alternative or to select a limited set of best alternatives. Next, Roy introduced the second goal as a rank-ordering of the alternatives from the best to the worse ones. The third objective encloses the classification/sort of the alternatives into predefined homogeneous groups. Finally, he presents the identification of the major distinguishing features of the alternatives and perform their description based on these features as the last goal.

In MCDA, problem types are defined in accordance to these objectives. Next, a more detailed explanation over the MCDA problematics [55] is described. Figure 1.1 provides an illustrative example of the problematics in MCDA.

- Description of the problem: it is necessary to define the set of alternatives, the set of criteria and/or attributes on which these alternatives will be estimated, and create the model of the problem. All the subsequent problems include the description as a first step. Indeed, it is treated separately just because it can be the only goal of decision aiding;

- Choice of the best alternative: during the decision aiding process the DM should find the smallest number of the best possible alternatives. The selection of one alternative or a set of best alternatives can be considered as another MCDA type of problem;

- Ranking the set of alternatives: corresponds to the DM's need of ordering the alternatives from the given set according to his or her preferences. An example of such a problem is the ranking of the students, from the best to the worst, according to their grades;

- Classification of alternatives: alternatives should be assigned into one or several classes from the set of possible ones. Such kind of problem appears, for instance, in diagnostic tasks (e.g., medical, technical, and financial). For example, a doctor diagnosis a disease according to the patient's symptoms.

The last three forms of decision making problems (choice, ranking, classification) lead to a specific result regarding the evaluation of the alternatives. Both choice and ranking are based on relative judgments, involving pair-wise comparisons between the alternatives. Consequently, the overall evaluation result has a relative form, depending on the alternatives being evaluated. For instance, an evaluation result of the form "product X is the best of its kind" is the outcome of relative judgments, and it may change if the set of products that are similar to product X is altered [77].

On the contrary, the classification problem is based on absolute judgments. In such cases, each alternative is assigned to a specific group on the basis of a pre-specified rule. The definition of this rule, usually, does not depend on the set of alternatives being evaluated. For instance, the evaluation result "product X does not meet the customer's needs" is based on absolute judgments, since it does not depend on the other products that are similar to product X. Of course, these judgments are not always absolute, since they are often defined within the general context characterizing the decision environment. For instance, under specific circumstances of the general economic and business environment a firm may fulfill the necessary requirements for its financing by a credit institution (these requirements are independent of the population of firms seeking financing). Nevertheless, as the economic and business conditions evolve, the financing requirements may change towards being stricter or more relaxed. Therefore, it is possible that the same firm is rejected credit under a different decision environment. Generally, despite any changes made in the classification rule used, this rule is always defined independently of the existing decision alternatives. This is the major distinguishing difference between the classification task and the problems of choice or ranking [77].

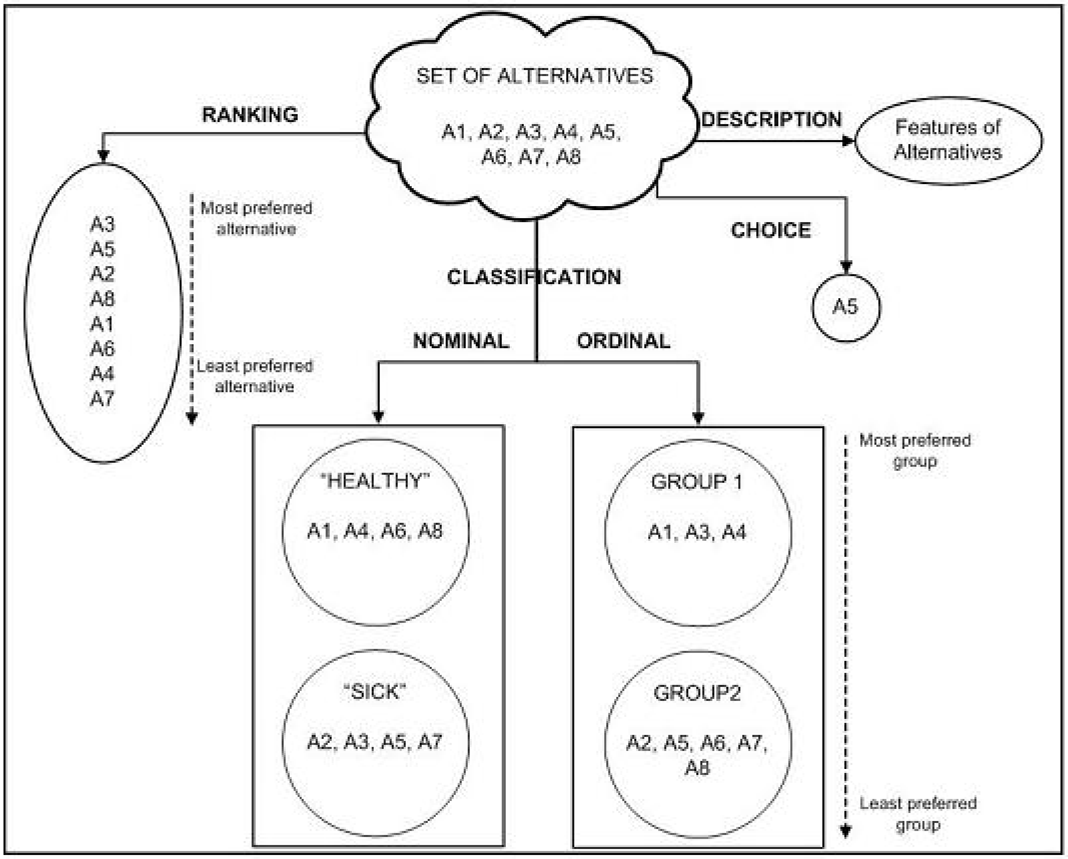


Fig. 1.1: MCDA Problematics.

1.2.6 Decision Aiding Process

The first definition of the decision aiding process is due to Simon [49]. It encloses concepts related to the cognitive activities of a DM that faces a problem for which no automatic pattern is available. The author emphasizes that different decision approaches in decision theory are intimately related to how the decision process is considered.

Bana and Costa et al. [19] present the three main phases of the decision aiding process:

- Structuring: during this phase, the analyst together with the DM should be engaged in structuring the problem area and identifying the main objectives. At this moment, also called as initialization step, it is necessary to identify, characterize and organize the aspects that should be relevant to the decision aiding process. In this context, the set of alternatives and the set of criteria, on which the alternatives are estimated, are selected. Then, the type of the problem situation is established, as well as the consequences of possible decisions. Yevseyeva [55] also includes the following additional information to be specified at this step: the set of classes for the classification problem and order of classes for the ordinal type of classification, and the requirements for the resulting information;

- Evaluation: the evaluation phase can be divided in the partial alternatives evaluation, considering each FPV (criterion), and a more generic evaluation considering all the small partial evaluations. Furthermore, the step of selecting a MCDA method is also performed here, considering from the set of possible ones with respect to the structure defined at the previous step and according to the type of the problem;

- Recommendation: at this point, many sensitive and robustness analysis are performed to examine if changes in the parameters, such as criteria weights and/or thresholds, can bring better results to the model. These modifications are usually made by the DM. However, this task may be difficult and cognitively loading for the DM. This difficulty has inspired the development of many special procedures that may simplify the parameters preference elicitation. This is a fundamental step that contributes to generate more knowledge over the domain problem, increasing the chances to obtain better results.

1.2.7 Multicriteria Methods

For Gomes et. al. [74], the multicriteria methods have been developed to support and guide, in different spaces, the decisors during the analysis and selection of the potential alternatives. The space of the decision variables consists in a set of feasible or non-feasible decisions for a certain problem. Some authors [3, 16] dismember the multicriteria methods in three family groups of approaches that refer to the preference modeling principles.

- Interactive judgment approach: The interactive methods, also known as Multi-Objective Decision Making, have mathematical programming techniques as a foundation, which encloses continuous set of alternatives with continuous solutions spaces [96]. The computational steps and the dialogue with the decisors alternate: the first computational step provides a initial solution that is presented to the decisors; on the other hand, they react supplying extra information about their preferences, allowing the construction of a new solution.

- Single synthesis criteria approach: From the American School, the fundamental principle of this approach is based on the following axiom: assume that any decisor is at least unconsciously maximizing some utility function during the decision aiding process. This function represents a global aggregation of alternatives utilities on each criterion. In other words, if the decisor is asked about his or her preferences, he or she will answer in a coherent way with some utility function (UT). The responsibility to estimate that function through some questionings to the decisor belongs to the analyst [4]. The preferences of the DM are expressed in the form of criteria weights. The weights in UT represent trade-offs between criteria. In order to represent uncertainties, UT uses probabilities [55].

  According to Yevseyeva [55], UT can work with different aggregation principles, which can be additive, multiplicative and distributional. This approach considers all the alternatives defined within a problem as comparable and independent. Since von Neumann and Morgenstern [63] who developed UT, many methods have appeared in the framework of this approach, and include Multiattribute Utility Theory (MAUT) [74, 101] and Analytical Hierarchical Process (AHP) [113].

- Outranking Approach: An outranking approach uses DM's preferences in order to compare pairs of alternatives. Based on the pairwise comparison, a complete or non-complete pre-order is built. Normally, the methods based on this approach use the concept of outranking relation (see Section 1.2.4). The main idea is to estimate the outranking degree of one alternative over another, that is, the alternative that loses in the higher number of criteria is considered worse than the other and vice-versa [51].

In comparison to UT, outranking methods can take into account incomparability of alternatives as well as have a non-compensatory character [55]. In practice, incompatibility situations can occur with the data's uncertainty and imprecision, and some implicit DM's characteristic. Two alternatives are considered incomparable if they are not indifferent to each other and it is not possible to say which of the two alternatives is better or worse. Compensatory models, such as UT, consider the trade-offs between criteria that are usually expressed by criteria weights. On the other hand, in the outranking approach, the weight is used to represent the relative importance of a criterion when compared to other criteria in terms of votes that support or oppose the assertion: one alternative is at least as good as another alternative [55].

The preference information for different outranking methods may be expressed in weights and through preference, indifference and veto thresholds. Currently, there are many MCDA methods developed according to the principles of this approach, which include ELECTRE [14, 15] and PROMETHEE [97] families of methods.

The above-mentioned approaches require some preferential information in the form of parameters of the model to be defined by the DM. However, in many situations, it may be difficult for the DM to understand the meaning of the parameters and define precise values for preferential parameters of a particular model. That is why, recently, two other MCDA directions have been intensively developed: preference elicitation procedures and rules inferring, also called here decision rule approach [23].

Preference elicitation procedures are indirect techniques that do not require the DM to express his preferences directly by entering the parameters of a model. Mousseau [116] defines the preference elicitation process as an activity that aims at making the DM's value system explicit through a model selected to model his/her preferences. Preference disaggregation approaches have been developed to elicit preferences in the form specified by the model (either based on the UT or the outranking approach) with one of the mathematical programming techniques. The literature presents a vast number of preference elicitation procedures over the UT theory and the outranking methods such as UTA [37], UTADIS [38, 77], ELECTRE TRI Assistant [120]. Usually, with these methods, there is no need for specifying any information (such as decisions made earlier by the DM), but, if some partial information about the parameters of the model (e.g., in the form of intervals of values or their distribution) is available, it can be used in the SMAA methods [55].

A decision rule approach is a technique that represents the DM's preferences in terms of "if ..., then ..." rules or decision trees [110]. This idea is related to the rough set theory and adapted from artificial intelligence, where a similar approach is used, for instance, in expert systems. The DM gives information in the form of examples of decisions, which requires relatively low cognitive effort and which is quite natural. The decision model is also expressed in a very natural way by decision rules. Many works have been developed in the framework of the decision rule approach, such as verbal decision analysis [91, 92], fuzzy sets [100], [72], and rough sets [110].

Having defined the context of MCDA and the main methodological approaches developed within this field, the subsequent analysis is focused on the MCDA methods proposed for addressing classification problems.

1.3 Objectives

The main objective of this work is to develop an approach to assist multicriteria nominal classification methods during the parameter estimation and prototype selection activities. Moreover, this research aims to achieve the following specific objectives:

- Assess the impact that the prototypes exert in two MCDA classification methods;
- Structure the integrated approach through the employment of different techniques;
- Develop a case study that applies the proposed approach over the Alzheimer's Disease early diagnosis detection.

1.4 Conclusion

In this chapter, a brief introduction was realized to specify some basic concepts of the MCDA theory used throughout this work. After that, the main MCDA relations, types of problems and the most widely-used methods were explained to provide a general overview of the field. In the next chapter, we provide the basis of the MCDA classification field.

Chapter 2

Multicriteria Decision Aiding Classification

There is a vast variety of methods for classification in MCDA. The diversity of methods is due to the different types of classification problems, the available information and requirements. This chapter covers a general outline over some MCDA theories and methods enclosing the classification problems.

2.1 The Classification Problem

A classification problem refers to the assignment of a group of alternatives to a set of predefined classes, also known as categories. During the last decades these problems have been tackled using a high variety of statistical and machine learning techniques. Recently, the MCDA area has also brought new methodologies and techniques to solve these problems. The main difference between the MCDA classification methods and others coming from related disciplines (for instance, artificial neural networks, Bayesian models, rule-based models, decision trees, etc.) [56] lies in the way that the MCDA methods incorporate the DM's preferences into the categorization process.

According to Doumpos and Zopounidis [77], in MCDA, the classification problems can be referred to as three specific terms that are often related to the classification problem: discrimination, classification and sorting. The first two terms are commonly used by statisticians as well as by scientists of the artificial intelligence field (neural networks, machine learning,etc.). The term "sorting" has been established by MCDA researchers. Although the three terms refer to the assignment of a set of alternatives into predefined groups, there is a notable difference between the kinds of problems that they describe. In particular, from the methodological point of view, the three terms describe two different kinds of problems [77].

The terms "discrimination" and "classification" refer to problems where the categories are defined in a nominal way. In such cases, the alternatives that are assigned to different categories present different features, without being possible to establish any kind of preference relation between them [77].

On the other hand, sorting refers to problems where the groups are defined in an ordinal way. A typical example of this sort of problems is the bankruptcy risk evaluation problem [106]. In this problem, there is an ordinal definition of the groups, since it is obvious that for a DM the healthy firms are in better situation than the bankrupt ones. Therefore, the definition of the groups does not only provide a simple description of the alternatives, but it also incorporates additional preferential information, which could be of interest to the decision making context [77]. For simplicity reasons, henceforth only the term "classification" will be used in this thesis. Moreover, this work will not focus on the selection of the MCDA method, which is being treated by many researchers [55, 108] as another MCDA problem.

In this context, the rest of this chapter categorizes the MCDA classification methods that are based on some of the most important MCDA theories. A deeper literature review over the MCDA classification methods can be found in [23, 55, 77, 122].

2.2 Outranking Relation

The basis of the outranking relation theory (ORT) have been set by Roy in the mid 1960s through the development of the ELECTRE (*ELimination Et Choix Traduisant la REalité*) family of methods [81]. Since then, the outranking methods have yielded a productive approach to many MCDA researchers. The main concept that underlies this approach is the pairwise comparison of all the feasible alternatives through the construction of binary relations, and their exploitation in order to obtain the final recommendations. These binary relations estimate the preference of an alternative over the same criterion.

According to Doumpos and Zopounidis [77], all ORT techniques operate in two major stages. The first stage involves the development of an outranking relation, whereas the second stage involves the exploitation of the outranking relation in order to perform the evaluation of the alternatives for choice, ranking, classification/sorting purposes.

Basically, the first stage encloses the definition of the preference, indifference and veto thresholds. These thresholds were introduced to define a fuzzy relation, and to stipulate an absence of the tradeoff between the criteria, which turns the ORT into a non-compensatory technique. In other words, a low performance of an alternative under a specific criterion is not compensated by the performance of the alternative under the remaining criteria.

After the development of the outranking relation, the next step is to employ the outranking relation for the different decision making purposes (choice, ranking, classification). During this stage, heuristic procedures are commonly employed to decide upon the evaluation of the alternatives on the basis of developed outranking relation. The most extensively used ORT techniques are the ELECTRE methods [15, 67, 59, 61, 81], as well as the PROMETHEE methods [68]. Doumpos and Zopounidis [77] stress that these two families of methods enclose many variants for different MCDA problems such as choice, ranking and classification. The next section presents further details on the ELECTRE TRI method.

2.2.1 ELECTRE TRI

ELECTRE TRI [121] is a member of the family of ELECTRE [105] methods. The ELECTRE methods are based on the outranking relation theory, and the ELECTRE TRI, specifically, was designed to solve ordinal classification problems. The objective of ELECTRE TRI is to assign a discrete set of alternatives into groups that are defined in an ordinal way.

Let's consider an ordinal classification problem where A is the finite set of alternatives to be classified, and F the set of n criteria, with $n \geq 1$, assigned into a set of ordered classes L predefined by the set of boundary alternatives $B = \{B^1, B^2, \ldots, B^s\}$. Each class is constrained by two (upper and lower) boundary alternatives. This method introduced the concept of reference profile as a fictitious alternative, which is a boundary between two consecutive groups. Boundary alternatives separate classes in such a way that the upper bound b_q of the class l_{q-1} is the lower bound of the class l_q ($q = 1, \ldots, s$). Changing values in at least one criterion moves the boundary alternative to a neighboring class. For solving a classification problem, the method estimates the outranking relation for each alternative $a_i \in A$ ($i = 1, \ldots, m$) to be classified with each boundary alternative b_q between classes l_{q-1} and l_q by calculating the outranking index. We assign the alternative a_i to the class l_q if it is preferred to the lower boundary alternative b_{q-1} of the class, and if the upper boundary alternative b_q of the class is preferred to this alternative.

Before the outranking index is computed, the DM needs to specify the set of alternatives and criteria that will take part into the classification process. In addition, the DM needs to specify the number of categories in a specific order according to his or her preferences and define the upper b_q and lower b_{q-1} boundary alternatives for each class l_q.

Besides this initial information that surrounds the problem, the ELECTRE TRI method requires the specification of the following thresholds for each criterion j: indifference q_j, preference p_j and veto v_j thresholds, weights w_j, and cutting level λ. The indifference threshold q_j represents the maximum allowed criterion difference between two alternatives such that they remain indifferent for the DM. The preference threshold p_j indicates the minimum allowed criterion difference between two alternatives such that one is preferred to the other under the criterion g_j. The veto threshold v_j represents the minimum allowed difference between two alternatives under the same criterion g_j that shows incomparability of these alternatives. It is important to observe that the thresholds should satisfy the following constraint: $v_j > p_j > q_j$. The weight w_j of a criterion g_j represents the importance of that criterion when compared to other criteria. Finally, the cutting level λ denotes the minimum value of the outranking index that is necessary to consider an outranking situation between two alternatives [55].

The outranking relation is checked through the concordance and non-discordance conditions, that consider all thresholds and weights defined by the DM. The first condition requires preference of the alternative a_i over the boundary alternative b_q in the majority of criteria and the second one demands the absence of strong opposition to the first condition in the minority of criteria. We compute two partial indices for each criterion: concordance $C_j(a_i, b_q)$ and $C_j(b_q, a_i)$, and discordance $D_j(a_i, b_q)$ and $D_j(b_q, a_i)$. They permit the calculation of the outranking indices $S(a_i, b_q)$ and $S(b_q, a_i)$. There are two possible assignment procedures (pessimistic and optimistic), which consist of a comparison of outranking indices to the cutting level λ. Next, the ELECTRE TRI algorithm is explained in detail.

The classification of the alternatives with ELECTRE TRI consists of two phases. The first phase involves the construction of an outranking relation that defines whether an alternative outranks a profile or not. The second phase exploits this relation for the assignment of alternatives into the classes.

Phase I. To define the outranking relation ($a_i S b_q$) for each alternative a_i and each boundary alternative b_q it is necessary to pass through the following steps:

Step I. Calculate the partial concordance indices $C_j(a_i, b_q)$ and $C_j(b_q, a_i)$ for each criterion g_j. When dealing with a maximization problem the criterion g_j presents an increasing direction of preference. The indices are computed as follows:

$$C_j(a_i,b_q)=\begin{cases}0, & \text{if } g_j(b_q)-g_j(a_i)\geq p_j(b_q)\\ \left(\frac{p_j(b_q)-g_j(b_q)+g_j(a_i)}{p_j(b_q)-q_j(b_q)}\right), & \text{if } g_j(b_q)-p_j(b_q)<g_j(a_i)\leq g_j(b_q)-q_j(b_q)\\ 1, & \text{if } g_j(b_q)-g_j(a_i)<q_j(b_q)\end{cases}\tag{2.1}$$

$$C_j(b_q,a_i)=\begin{cases}0, & \text{if } g_j(a_i)-g_j(b_q)\geq p_j(a_i)\\ \left(\frac{p_j(a_i)-g_j(a_i)+g_j(b_q)}{p_j(a_i)-q_j(a_i)}\right), & \text{if } g_j(a_i)-p_j(a_i)<g_j(b_q)\leq g_j(a_i)-q_j(a_i)\\ 1, & \text{if } g_j(a_i)-g_j(b_q)<q_j(a_i)\end{cases}\tag{2.2}$$

Step II. Define the overall concordance indices as an aggregation of partial concordance indices.

$$C(a_i,b_q)=\frac{\sum_{j=1}^{n}w_jC_j(a_i,b_q)}{\sum_{j=1}^{n}w_j},\tag{2.3}$$

$$C(b_q,a_i)=\frac{\sum_{j=1}^{n}w_jC_j(b_q,a_i)}{\sum_{j=1}^{n}w_j}\tag{2.4}$$

Step III. The third step calculates the partial discordance indices $D_j(a_i,b_q)$ and $D_j(b_q,a_i)$ for each criterion g_j. This index is computed according to the increasing direction of preference as follows:

$$D_j(a_i,b_q)=\begin{cases}0, & \text{if } g_j(b_q)-g_j(a_i)<p_j(b_q)\\ \left(\frac{g_j(b_q)-g_j(a_i)-p_j(b_q)}{v_j(b_q)-p_j(b_q)}\right), & \text{if } g_j(b_q)-v_j(b_q)<g_j(a_i)\leq g_j(b_q)-p_j(b_q)\\ 1, & \text{if } g_j(b_q)-g_j(a_i)\geq v_j(b_q)\end{cases}\tag{2.5}$$

$$D_j(b_q,a_i)=\begin{cases}0, & \text{if } g_j(a_i)-g_j(b_q)<p_j(a_i)\\ \left(\frac{g_j(a_i)-g_j(b_q)-p_j(a_i)}{v_j(a_i)-p_j(a_i)}\right), & \text{if } g_j(a_i)-v_j(a_i)<g_j(b_q)\leq g_j(a_i)-p_j(a_i)\\ 1, & \text{if } g_j(a_i)-g_j(b_q)\geq v_j(a_i)\end{cases}\tag{2.6}$$

Step IV. Calculate the outranking index $S(a_i,b_q)$ that demonstrates the outranking credibility of a_i over b_q, knowing that $S(a_i,b_q)\in$ [0,1], as follows:

$$S(a_i,b_q)=C(a_i,b_q)\prod_{j=1}^{n}\frac{1-D_j(a_i,b_q)}{1-C_j(a_i,b_q)},\tag{2.7}$$

where j = (1, . . . , n) and $D_j(a_i,b_q)$ > $C_j(a_i,b_q)$.

Step V. During this step the DM defines the cutting threshold λ. This value is, usually, defined over the [0.5, 1] interval that indicates the minimal value of outranking indices accepted for outranking two alternatives. The value of the outranking index $S(a_i,b_q)$ is compared to the cutting level λ. The method uses that comparison to specify the preference relation between the alternatives.

- $S(a_i,b_q)\geq\lambda$ and $S(b_q,a_i)\geq\lambda\Longrightarrow a_iIb_q$, then the alternatives a_i and b_q are indifferent;

- $S(a_i, b_q) \geq \lambda$ and $S(b_q, a_i) < \lambda \Longrightarrow a_i P b_q$ or $a_i Q b_q$, then the alternatives a_i is strongly or weakly preferred to b_q;
- $S(a_i, b_q) < \lambda$ and $S(b_q, a_i) \geq \lambda \Longrightarrow b_q P a_i$ or $b_q Q a_i$, then the boundary reference alternative b_q is strongly or weakly preferred to the alternative a_i;
- $S(a_i, b_q) < \lambda$ and $S(b_q, a_i) < \lambda \Longrightarrow a_i J b_q$, then the alternatives a_i and b_q are incomparable.

Phase II. After the definition of the preference outranking relation the DM will exploit it by choosing the assignment procedure: pessimistic, optimistic or both. Then, it is necessary to compare the outranking indices for each pair composed of the alternative a_i to be classified and each boundary alternative b_q to the cutting level λ.

If the DM chooses the pessimistic procedure, he or she will start to compare the alternative a_i to the lower bound b_{q-1} of the highest class l_q and continue in a decreasing order until it finds such a lower bound b_{q-1} that is outranked by the alternative a_i ($a_i S b_{q-1}$). For that, the estimation of the outranking relation $a_i S b_{q-1}$ is performed, and the outranking index $S(a_i, b_{q-1})$ is calculated. Then, the outranking index between the alternative a_i and the upper bound b_q of the class l_q is calculated $S(a_i, b_q)$. Finally the method assigns the alternative a_i to the class l_q if $S(a_i, b_{q-1}) \geq \lambda$ and $S(a_i, b_q) < \lambda$.

On the other hand, the optimistic assignment procedure assumes that we begin to compare the alternative a_i to the upper bound b_q of the lowest class l_q and maintain in an increasing order until it finds such an upper bound b_q that has strict preferences over the alternative a_i ($b_q P a_i$). Then, the outranking index between the alternative a_i to be classified and the lower boundary alternative b_{q-1} of the same class l_q is calculated $S(a_i, b_{q-1})$. Lastly, the alternative is assigned to the class l_q if $S(a_i, b_{q-1}) \geq \lambda$ and $S(a_i, b_q) < \lambda$.

If the classification is performed taking into consideration one of the above-mentioned procedures it is considered unambiguous. On the other hand, if the classification is performed considering both of the procedures, the same alternative can be assigned to two different classes. In such cases, this ambiguity is resolved either by changing the assignment procedure or with the assistance of the DM [81].

In ELECTRE TRI, it is necessary to define a large number of parameters: boundary alternatives, thresholds and the weights for each criterion as well as the cutting level. In most cases the DM presents some difficulty in defining precise values for all that preferential information due to various reasons. Belacel [86] highlights some important factors that can turn this process complicated for the DM. For example, the data considered in the model might be imprecise or uncertain. Another issue is that the experts may have only a vague understanding of parameters and their opinions and judgments can evolve during the elicitation process. Furthermore, the thresholds elicitation requires from the DM an ability to understand the meaning of each parameter and to operate with differences on criteria values, which may be cognitively difficult for the DM. That is why many researches consider the idea of inferring preferential parameters from examples very attractive.

In this context, it may be easier for the DM to provide the examples of ready decisions that were made in the past. This fact was considered by researchers and developers of preference disaggregation approach (PDA) models (see Section 2.3). This approach does not requires the DM to express his or her preferences with specific model parameters but to provide the analyst with ready decisions [38]. In practice, these models have been widely-used for eliciting preferential information such as: weights [118], boundary alternatives [86], and veto thresholds [117]. The main features, advantages and disadvantages of this methodology are discussed in the next section.

The ELECTRE TRI method has successfully been applied to a large number of practical problems. In what follows some examples of the ELECTRE TRI employment in a vast range of applications are given: human resource management [73], greenhouse gas emissions [36], assessment of land-use stability [41], customer satisfaction [50], and in software evaluation [39].

2.3 Preference Disaggregation

Trying to facilitate the parameter elicitation procedure, several researches [38, 86] investigated some other ways to simplify that process. MCDA theories such as UT and ORT are related to the modeling and representation of the DM's preferential system in a pre-specified mathematical model (utility function or outranking relation). On the other hand, the focus in PDA is on the development of a methodological framework, which can be used to evaluate the decisions taken by the DM so that an appropriate model can be constructed representing the DM's preferences. Indeed, PDA does not ask the DM to provide any sort of information on how the decisions are taken. As a matter of fact, PDA investigates the relationship between the decision factors (evaluation criteria) and the actual decisions [77].

PDA presents its foundations over the complexity of eliciting the DM's preferences. This difficulty is due to the time constraints and the unwillingness of the DMs to participate actively in such an interactive elicitation/decision aiding process. The PDA theory facilitates this process by letting the DMs to express their actual decisions, without providing any other information on how these decisions are taken (e.g., significance of criteria) [77].

In a general way, the PDA paradigm is similar to the regression framework used extensively in other disciplines, such as statistics, econometrics and artificial intelligence for model development purposes. In fact, the foundations of PDA have been set by operations researchers, which have used robust mathematical programming models in an attempt to develop non-parametric regression-based techniques [77].

Jacquet-Lagréze and J. Siskos developed the first PDA method that enclosed the MCDA ranking type of problems. During the first versions of the UTilités Additives (UTA) method a weighted sum preference model was defined, for which parameters were estimated through a linear programming. During the last decades the method evolved and more complex models have been developed. The work of Jacquet-Lagréze and J. Siskos [38] demonstrated a comprehensive review of this MCDA approach and the developments made over the past two decades. Next, the MCDA classification method developed in the framework of PDA is considered.

2.3.1 UTADIS

In the MCDA field, the first classification method employed was the UTADIS (UTilités Additives DIScriminantes). This method was developed based on the UTA method and discriminant analysis [58, 77], and enclosed the ordinal classification type of problems. As other PDA methods, UTADIS reduces the cognitive loading of the DM by not requiring him or her to express his or her preferences into a set of model parameters. In addition, all necessary information about assigning alternatives to each class is elicited from the DM. After that, this information is used to stipulate the parameters through a mathematical programming model.

Initially, the UTADIS method assumes that the set of assignment examples $A = \{a_1, \dots, a_m\}$ is available and will be estimated on the set of criteria $G = \{g_1, \dots, g_n\}$ that may have verbal or numerical values and be defined by a linear or nonlinear criterion function. For each discrete criterion g_j from the set G, the scale of values $S(g_j) = \{g_{j1}, \dots, g_{jt}\}$ (where t is the number of values on the scale of criterion g_j) is defined. For the continuous type of criterion, the function is available and also its minimal and maximal values [g_j^{min} , g_j^{max}] $\in S(g_j)$.

UTADIS allows the criteria functions $u_j(g_j(a_i))$, where $a_i \in A$ and $g_j \in G$, to have different types of scales (numerical and verbal) with different measures to be reduced to a new scale that ranges from 0 to 1. These functions are linearly or nonlinearly described over the scales of criteria. This scale represents the utility for the DM of each criterion's value. The form of the marginal utility functions depends upon the DM's preferential system (judgment policy) [77]. Commonly, there are three different forms: the concave form of the utility function indicates that the DM considers as quite significant small deviations from the worst performance. This corresponds to a riskaverse attitude (e.g., the DM is satisfied with "acceptable" performance and he or she does not necessarily seek alternatives of top performance), and it is represented as a concave function. On the contrary, the riskprone behavior corresponds to the DM's interest for alternatives of top performance, that is represented with a convex function. Finally, the linear marginal utility function indicates a riskneutral behavior.

Formally, the criteria aggregation model is expressed as an additive weighted sum utility function:

$$u(a_i) = \sum_{j=1}^{n} w_j u_j(g_j(a_i)), \tag{2.8}$$

where $u(a_i)$ is the utility of the alternative $a_i \in A$, and $u_j(g_j(a_i))$ is the utility of the alternative a_i on the function of criterion g_j; moreover, w_j is the weight of the criterion g_j.

In this context, the UTADIS method is briefly defined by six steps as follows:

Step I. Approximate each nonlinear criterion function with piecewise linear functions as follows. It is assumed that the nonlinear function of the criterion g_j takes values on some interval [g_j^{min} , g_j^{max}] $\in S(g_j)$, which is divided into z linear subintervals [g_j^0 , g_j^1[,…,[g_j^{z-1}, g_j^z]]]. In order to classify an alternative, it is necessary to know all the utilities of the breakpoints $u_j(g_j^0) \leq \dots u_j(g_j^z)$ for all subintervals [g_j^k, g_j^{k+1}], where (k = 0, …, z) and $g_j \in G$. Indeed, this set of utilities of breakpoints needs to be defined through a PDA in UTADIS.

Step II. The general performance of an alternative is estimated by aggregating the performance of the alternative on each criterion, taking into account the criteria weights. Then, each alternative should be classified by comparison of its global utility to the utility thresholds of the classes according to the following rule: the alternative is assigned into the class l_q (q = 1, . . . , s) if its utility is bigger than the lower utility threshold of the class and is smaller than the upper threshold of the class:

$$\begin{cases} \text{if} \quad 0 \leq u(a_i) < u_1, \quad \text{then } a_i \in \mathrm{l}_1 \\ \text{if} \quad u_{q-1} \leq u(a_i) < u_q, \quad \text{then } a_i \in \mathrm{l}_q \\ \text{if} \quad u_{s-1} \leq u(a_i) \leq 1, \quad \text{then } a_i \in \mathrm{l}_s \end{cases} \tag{2.9}$$

where the set $\{u_1$, … , $u_{s-1}\}$ defines utility thresholds between neighboring ordered classes $\{l_1$,

$\ldots, l_s\}$, respectively. In order to be able to perform the classification according to this set of rules, it is necessary to know the utility thresholds between all classes. This set of parameters will be defined through PDA in UTADIS.

Step III. Define the objective function. For the UTADIS original method, the goal is to create a classification model that reassigns the set of test samples $a_i \in A$ as closely as possible to the set of ordered classes. In this context, the misclassification error rate has to be minimized.

Step IV. Apply the slack variables into the classification rules. The first variable σ_i^+ is introduced, because, in order to classify the alternative a_i into the class l_q correctly, it is necessary to augment its global utility $u_q - u(a_i)$. On the other hand, the slack variable σ_i^- defines misclassification of the alternative a_i such that in order to assign it correctly into the class l_q its global utility should be decreased by $u(a_i) - u_{q-1}$.

Step V. This step has as objective to define the constraints for the problem. In addition to the constraints defined by the classification rules (step IV), the constraints for the weights to be nonnegative and normalized should be taken into account.

Step VI. Finally, the last step creates an optimization problem and solves it through a mathematical programming model, taking into account the goal of the UTADIS method, that is the minimization of classification errors of the assignment examples $a_i \in A$ provided by the DM. The constraints of this mathematical programming model have a nonlinear form and solving this nonlinear programming problem could be challenging [55].

Two heuristics have been introduced in order to overcome this problem [77, 86]. The first one assumes that nonlinear criterion functions are divided in subintervals in such a way that there is at least one assignment example belonging to each subinterval. The other alternative takes into account the distribution of assignment examples from different classes on each criterion's scale [55].

The UTADIS application success led many researchers to investigate the method's properties. Doumpos and Zopounidis compared the UTADIS performance to some statistical techniques and MCDA methods [21, 77]. The authors also investigated several issues regarding the impact of the parameters involved in the model development process on the performance and the stability of the developed models [78]. Mousseau et al. [119] verified the inclusion of the size of each class into the UTADIS method.

Most of UTADIS advantages rely on the fact that it facilitates the classification with the set of assignment alternatives, by not making the DM to provide complex input parameters. Another advantage is the use of a straightforward and understandable criteria function throughout the method. However, Oral and Kettani [80] have criticized the method for not taking into consideration the interaction between criteria. Another negative issue related to PDA methods was evidenced by Belacel [86]. The author considers that it is very hard to solve such mathematical programming models using the classical methods such as gradient algorithms, generalized reduced method and interior-point algorithms due to its computational complexity. The UTADIS method evolved along the years, and several versions of UTADIS (UTADIS I, II, III) have been released [22, 76].

2.4 Verbal Decision Analysis

The Verbal Decision Analysis (VDA) methods are essentially based on the verbal description of the decision problems in a natural language, which sometimes is the only way for the DM's to express their thoughts. In general, these methods have as objective to be transparent to the DM, by using qualitative verbal information to the evaluation of criteria and alternatives that are expressed in an ordinal scale.

One of the main features of the VDA methods is the ability to operate with qualitative variables without direct conversion into numerical values; thus they do not make use of quantitative operations such as addition or multiplication. Instead of applying complex operations, these methods combine simple operations and procedures that can be easily understood by the DM [53]. According to Yevseyeva [55], the idea behind the VDA methods is to take into consideration the cognitive aspects of human behavior. These methods assume elicitation of DM's preferences in a natural language that sometimes can be the only way for the DM to express his or her thoughts.

Most MCDA methods are not suitable for working purely with verbal descriptions; indeed, they need to convert them to produce the results in the numerical form. When it is not possible to employ that conversion, the DM has to establish all the preferential information. However, there are some situations that the DM is not able to provide precise information on the criteria or that information is lost during the conversion.

Larichev et al. [89, 92] developed the VDA methods with the focus of aiding in the decision making of unstructured problems. These methods exempt the DM effort to obtain detailed quantitative estimatives, which usually are uncertain and imprecise for these problems. Moreover, for Newell et al. [4], it is difficult for this type of problem to provide numerical evaluations on criteria as well as to estimate dependencies between them. In such problems, it is common to only have a relative definition of scales of criteria, because, in most cases, the absolute definition is not at the DM's disposal. For instance, it may be known that one value is better than another, but the information about how much better may be absent. In such problems, criteria, instead of having objective scales, may only have subjective scales that are based on the DM's preferences, his or her knowledge, experience and intuition [55].

Larichev and Moshkovich [92] describe the following routines that, in general, are present in all VDA methods:

- During the decision making process the information is not complete, turning the alternatives undefined;
- The decision is guided by the DM's intuition and conviction about the occurrence of possible events. In such cases, from the DM's subjective preferences, it can only be elaborated generic estimatives over the alternatives;
- Iterative and interactive dialog between the DM and the algorithm, in the framework of which the DM has to classify some alternatives, but not all;
- The evaluation of the alternatives is performed in a qualitative and natural language manner, which is based on ordinal scales;
- The problem description can influence the way the DM interacts with the problem.

The VDA classification process is iterative and requires an active DM participation. At each iteration, the DM assigns one alternative selected by the VDA method to a specific class. Based on the DM's answer, other alternatives may be automatically classified by the method. The process is repeated until each alternative is assigned to exactly one class. In case of a contradiction, the DM needs to rethink his or her answers. At the end of the classification, when each alternative is classified, the set of rules that describes classes can be obtained. Based on this set of rules, the DM can get an explanation about the appearance of each alternative in the class [55].

Larichev and his colleagues have dedicated their research to develop several methods based on the VDA principles to support different types of MCDA problems [87, 89, 90, 91, 92]. Furthermore, ZAPROS-LM [92] creates a partial rank-ordering of alternatives according to their importance for the DM. A practical study containing the application of this methodology can be found in [54]. PAired COMpensation (PACOM) [92] works with the choice problematic by selecting the best alternative from the set of given alternatives. Recently, the ORdinal CLASSification (ORCLASS) [92], the Subset Alternative Classification (SAC) [89] and CYCLE [88] methods were developed to classify a set of alternatives into ordered classes.

2.5 Fuzzy Sets

Commonly, decision making is involved with ambiguous, uncertain and fuzzy issues. These concepts are incorporated by the use of subjective expressions such as "majority", "generally", "often", etc., that brings the fuzzy concept into the real context of applications. The fuzzy sets theory was developed and adapted to the MCDA context by Zadeh [72, 100], and since then, many MCDA methods started to treat the uncertainty in a different manner.

In relation to the classification methods, it can be stated that there is a partial knowledge, which is not well defined, in the assignment procedure of an alternative to a specific class. This way, the following proposition "alternative a belongs to the set A" can be partially true or false in the fuzzy sets theory. To model this sort of fuzzy judgments, the fuzzy sets theory introduced the membership functions. The objective of these functions is to define the membership degree of an alternative to belong to a fuzzy set. These functions produce results with intervals that range from 0 to 1, with the maximum value indicating the truth of a judgment and the minimum value indicating that a proposition is false. For the classification problem, a membership function shows the degree to which an alternative estimated on a criterion or attribute belongs to a class.

Figure 2.1 defines the general way to present a fuzzy membership function μ for the proposition A_j^q: "a_i belongs to the class l_q according to the criterion or attribute g_j". In addition, the figure also shows the complementary part for the negation of the proposition $\neg \mathrm{A}_j^q$:"a_i does not belong to the class l_q according to the criterion or attribute g_j".

It is assumed that the DM is able to provide the criteria and attributes values and define such of them that support and/or oppose assignment of alternatives to each class. To provide a conclusion regarding the membership of an alternative into a fuzzy set, a procedure that aggregates the partial membership degrees for each individual attribute is applied. This fuzzy rules aggregation behavior may be defined as follows:

$$\text{If } A_1^q \wedge \ldots \wedge A_n^q, \quad \text{then} \quad a_i \in l_q, \tag{2.10}$$

where the fuzzy value of the membership function A_j^q corresponds to each value of criterion or attribute g_j when assigning the alternative a_i to the class l_q.

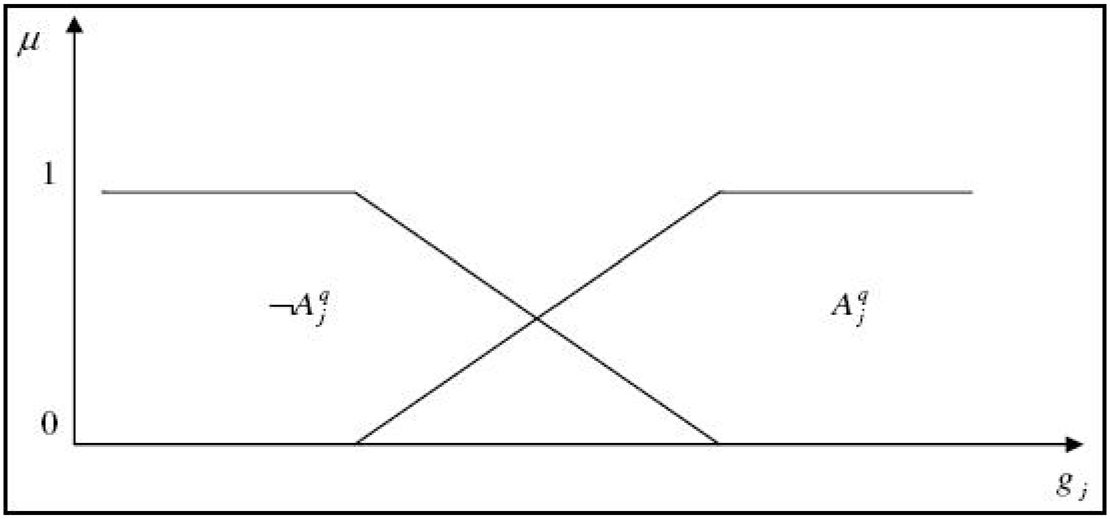


Fig. 2.1: Partial membership function for fuzzy set [55].

Many studies concerning the MCDA fuzzy sets literature have been developed, as this theory evolves and appears as a promising way of modeling the uncertainty related to the human preferences. Another positive aspect is related to the ease and simplicity in applying the fuzzy functions, even though complicated membership functions can be applied as it can be seen in [77]. Fuzzy sets have been extensively used to address a vast variety of real-world problems from several areas. Furthermore, the combination of the fuzzy sets theory with other disciplines has been subject of studies of several researchers. For instance, the neural networks [27], expert systems [82], mathematical programming [52] and MCDA [28, 43, 60, 62, 71, 109] areas already incorporated those concepts.

Despite its vast range of application areas, some authors [52, 114] evidenced practical issues involving the implementation of the fuzzy sets. A common problem is related to the choice of the membership function. In order to do so, the DM has to be apt to estimate the importance of the criterion or attribute values to support the alternatives classification procedure into each class.

2.6 Machine Learning

During the last two decades Machine Learning evolved as a major discipline within the artificial intelligence discipline. Its objective is to describe and analyze the computational procedures required to extract and organize knowledge from the existing experience [77].

Within the different learning paradigms [124], this section have its focus on the instance-based learning paradigm that encloses a particular algorithm known as k-NN, which is similar to some MCDA classification techniques.

Instance based learning (IBL) algorithms [35] are a subset of the machine learning algorithms that uses instances (criteria vectors) from a stored reference set, which acts as a training set, as a means of generalizing the classes of new instances. These algorithms face the challenge of finding the ideal number of instances to save for generalizing. Wilson and Martinez [30] and Brighton and Mellish [46] provide an in depth survey of IBL algorithms of the past and their development through the years.

2.6.1 K-NN

In the Machine Learning field, the k-NN algorithm [125] is amongst the simplest and most-widely known classification methods. Instead of using a mathematical model, the classification procedure is based on the k most similar instances within a training set. Normally, the algorithm uses the Euclidean distance as a similarity measure, but many others techniques can be found in the literature [98].

Differently from many other machine learning methods, the k-NN training phase consists only of storing the feature (criteria) vectors and class labels of the training samples. Moreover, the test sample (alternative to be classified) is represented as a vector in the feature space.

The first step of the k-NN method is to compute the distance of the alternative to be classified and all the training set. After that, the algorithm applies a specific technique to assign the new alternative to a category. There is a vast number of techniques to classify the new vector to a particular class, one of the most used ways is to predict the new vector through a majority vote procedure, which assigns the alternative to the most common class amongst the k nearest neighbors. A major issue presented by this procedure is that classes with the higher number of alternatives tend to dominate the prediction, as they tend to appear (win) in the k nearest neighbors. To overcome this problem, many researchers compute the distance of the k neighbors to the test sample and assign it based on those distances.

Another problem that has been widely investigated and it is concerned to the k-NN is the difficulty that it has to deal with large datasets due to the computational costs involved. To solve that issue, some research was developed to apply instance selection to reduce the number of alternatives of a dataset. For the k-NN, methods such as CNN [95], ENN [31], VSM [29], and Multedit [94] have been successfully applied over the instance selection problem.

2.7 Nominal Classification

In the nominal classification or nominal sorting problems the alternatives are assigned to classes that are not ordered based on the employment of a similarity measure. In order to provide this index, the nominal classification methods can establish the outranking make use of or other preference relation technique.

Among the different MCDA classification approaches, this work focus on the investigation of two nominal classification methods. In this section, we describe these nominal classification methods, recently developed, which differ in the way that the similarity measure is applied.

Analytically, the models were defined in the following way: Let A be the finite set of alternatives, B the set of prototypes, F the set of n features (criteria), with $n \geq 1$, C is the set of categories of a problem where $C = \{C^1, C^2, \ldots, C^K\}$ and $K > 1$, and $B^h = \{b_p^h | p = 1, \ldots, L_h$ and $h = 1, ..., K\}$ the set of prototypes of the category C^h, where b_p^h represents the p-th prototype of the category C^h and L_h the number of prototypes of this category. Each alternative in A and B is characterized by a feature vector $\overline{g}$ containing its feature values for all n criteria in F. Each alternative $a \in A$ is compared with each prototype b_p^h under each criterion j.

2.7.1 gMCDA classification model

The first MCDA classification method that we have chosen to investigate was that proposed by Goletsis et al. [123] (referred to hereafter as *gMCDA classifier*). This method makes use of prototypes to serve as references against the new alternatives compared (matched) with it. One distinctive aspect of this scheme with respect to other MCDA-based classifiers is that it presents less control parameters to be adjusted (only some thresholds and criteria weights). In what follows, we provide further details of gMCDA.

As described by Goletsis et al. [123], during the comparison between an alternative and a prototype the first thing to be computed is the Similarity Index $(SI_j(a, b_p^h))$. This index is calculated for each criterion, and its objective is to model the criteria into a five zone similarity index. In order to compute this index, two thresholds must be specified.

The first threshold that needs to be specified is the similarity threshold, q_j, which represents the maximum allowed criterion difference $|g_j(a) - g_j(b_p^h)|$ between the alternatives and the prototypes. Using this, the alternatives can be judged similar under a specific criterion.

The second threshold used by the $(SI_j(a, b_p^h))$ computation is the dissimilarity threshold, p_j, representing the minimum allowed criterion difference between an alternative a and prototype b_p^h. This threshold needs to be defined in order to consider the criteria totally dissimilar.

The Similarity Index(SI) $(SI_j(a, b_p^h))$ is computed as described below:

$$SI_j(a, b_p^h) = \begin{cases} 1, & \text{if } |g_j(a) - g_j(b_p^h)| \leq q_j \\ \left(\frac{|g_j(a) - g_j(b_p^h)| - p_j}{q_j - p_j}\right), & \text{if } q_j < |g_j(a) - g_j(b_p^h)| < p_j \\ 0, & \text{if } |g_j(a) - g_j(b_p^h)| \geq p_j \end{cases} \quad (2.11)$$

After the computation of the similarity index, the next step is to compute the concordance index (CI). This index indicates the overall similarity concordance of an alternative a with a prototype b_p^h. This index is computed as follows:

$$CI(a, b_p^h) = \sum_j w_j \, SI_j(a, b_p^h), \quad (2.12)$$

where w_j is the weight of a specific criterion and $\sum_j w_j = 1$.

Each alternative will have its CI computed for all prototypes of all classes. After that, the next step is the computation of the membership degree (MD) of an alternative a to a category C^h. The best CI of a to all prototypes of C^h is given to the Membership Degree (MD). The MD is computed as follows:

$$MD(a, C^h) = max\{CI(a, b_1^h), \ldots, CI(a, b_{L_h}^h)\} \ . \quad (2.13)$$

Finally, the last step is the assignment of the alternative a to the category $C(a)$ with the maximum MD calculated in relation to all the groups of prototypes. The formula is presented below.

$$C(a) = arg \ max_h \, MD(a, C^h) \ . \quad (2.14)$$

The gMCDA method was first applied to the ischemic beat classification problem [123]. According to the authors [123] the main difficulty encountered when applying their method is the specification of weights and thresholds p_j and q_j. Aiming to achieve better performances and trying to obtain an automated beat classification, the authors have incorporated a genetic algorithm for the adjustment of the parameters of the multicriteria method. Futher information concerning the implementation of this algorithm can be found in [123].

2.7.2 PROAFTN classification model

The other MCDA classification method implemented also makes use of prototypes to serve as references against which the new alternatives are compared (matched) with. Differently from the gMCDA, the PROAFTN method [86] substitutes the similarity relation by the outranking relation, through the calculation of an indifference index based on an alternative and a reference profile (prototype).

The assignment procedure consists of calculating the degree of membership of each alternative to be assigned to each class based on the fuzzy indifference relation between this alternative and each prototype. Belacel [86] defines the principle in the following way: when the alternative a is judged indifferent to a prototype b_p^h according to the majority of criteria (majority principle) and there is no criterion which uses its veto against the affirrmation "a is indifferent to b_p^h" (minority respect principle), the action a is considered overall as indifferent to a prototype b_p^h. In order to calculate the fuzzy indifference relations it is necessary to build the partial indifference indices using the concordance and non-discordance concepts to aggregate them. As in the gMCDA algorithm the alternative will be assigned into the class with the maximal membership degree value [86].

In general, the prototype scores are given by intervals, so for each criterion g_j, we associate to each prototype b_p^h the interval $[S_j^1(b_p^h), S_j^2(b_p^h)]$, with $S_j^2(b_p^h) \geq$S$_j^1(b_p^h)$. The comprehensive indifference index is determined by aggregating the partial indifference indices. These indices indicate if the action a is indifferent or not to a prototype b_p^h according to a criterion g_j. The partial indifference relation is given as follows:

$$aI_jb_p^h \Longleftrightarrow g_j(a) \in [S_j^1(b_p^h), S_j^2(b_p^h)], \tag{2.15}$$

If the value of the alternative a according to the criterion g_j is equal to $S_j^1(b_p^h)$ or to $S_j^2(b_p^h)$, the alternative a will be indifferent to prototype b_p^h according to Equation 2.15. However, considering the imperfection and imprecision of the data, the alternative a on the criterion g_j can be assessed as: $g_j(a) = S_j^1(b_p^h) - \epsilon$ or $g_j(a) = S_j^2(b_p^h) - \epsilon$, where ϵ is a real number, which take very small values [86]. In this case, the application of the Equation 2.15 leads to transform the indifference situation into a non-indifference situation between the alternative a and prototype b_p^h according to criterion g_j, despite the fact that variation is not significant. In order to remedy this inconvenience, Belacel [86] introduced two discrimination thresholds $d_j^-(b_p^h) \geq 0$ and $d_j^+(b_p^h) \geq 0$, which correspond, respectively to two functions of $S_j^1(b_p^h)$ and $S_j^2(b_p^h)$.

Formally, three comparative situations between the action a and prototype b_p^h according to criterion g_j are obtained using the two discrimination thresholds:

- If $S_j^1(b_p^h) \leq$g$_j(a) \leq$S$_j^2(b_p^h)$, then a is clearly indifferent to b_p^h;
- If $[g_j(a) \leq$S$_j^1(b_p^h) - d_j^-(b_p^h)]$ or $[g_j(a) \geq$S$_j^2(b_p^h) + d_j^+(b_p^h)]$, then a is not indifferent to b_p^h;
- If $[S_j^1(b_p^h) - d_j^-(b_p^h) < g_j(a) < S_j^1(b_p^h)]$ or $[S_j^2(b_p^h) < g_j(a) < S_j^2(b_p^h) + d_j^+(b_p^h)$, then there is a weak indifference between a and b_p^h.

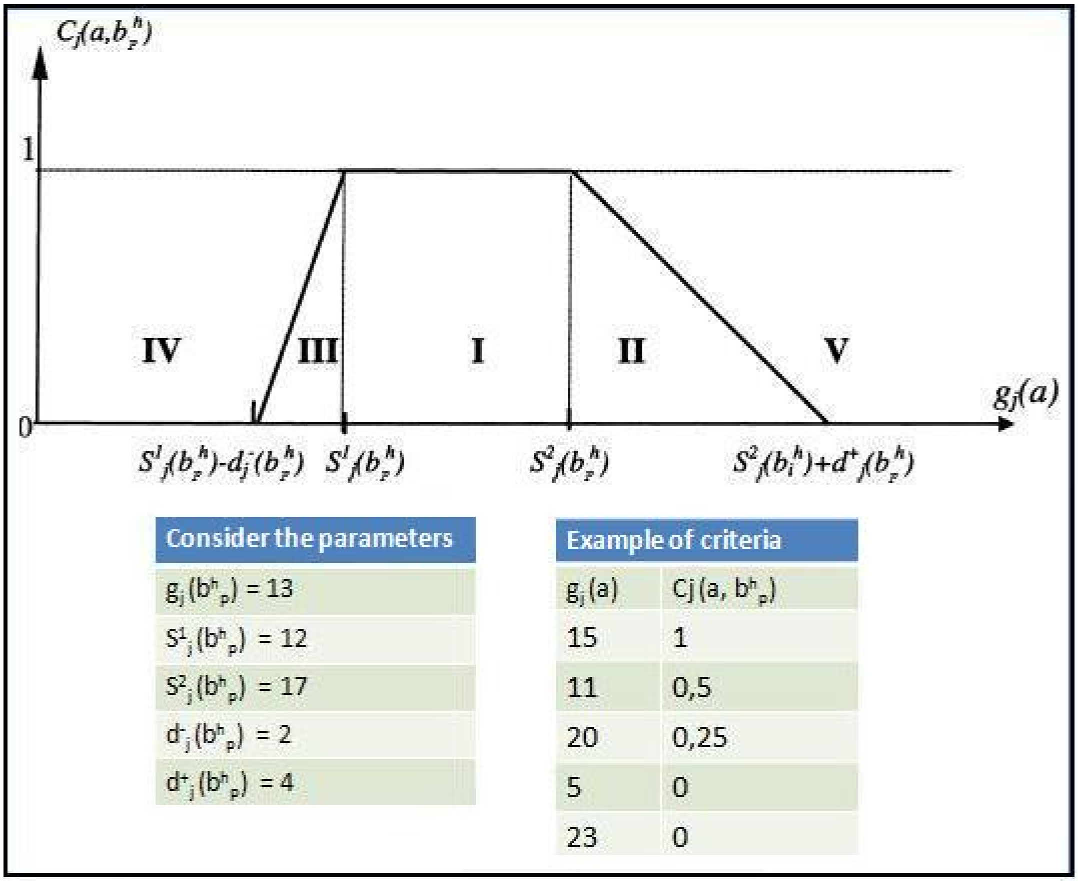
For each alternative a from the set of alternatives A to be classified and each reference alternative b_p^h of the class C^h, the partial concordance index on the criterion j is computed as follows. Figure 2.2 shows a more didatic example of the partial concordance index computation.

$$C_j(a, b_p^h) = min\{C_j^-(a, b_p^h), C_j^+(a, b_p^h)\}, \tag{2.16}$$

where,

$$C_j^-(a, b_p^h) = \frac{d_j^-(b_p^h) - min\{S_j^1(b_p^h) - g_j(a), d_j^-(b_p^h)\}}{d_j^-(b_p^h) - min\{S_j^1(b_p^h) - g_j(a), 0\}} \tag{2.17}$$

$$C_j^+(a, b_p^h) = \frac{d_j^+(b_p^h) - min\{g_j(a) - S_j^2(b_p^h), d_j^+(b_p^h)\}}{d_j^+(b_p^h) - min\{g_j(a) - S_j^2(b_p^h), 0\}} \tag{2.18}$$



Consider the parameters
$g_j(b^h_p) = 13$
$S^1_j(b^h_p) = 12$
$S^2_j(b^h_p) = 17$
$d^-_j(b^h_p) = 2$
$d^+_j(b^h_p) = 4$

Example of criteria	
$g_j(a)$	$Cj(a, b^h_p)$
15	1
11	0,5
20	0,25
5	0
23	0

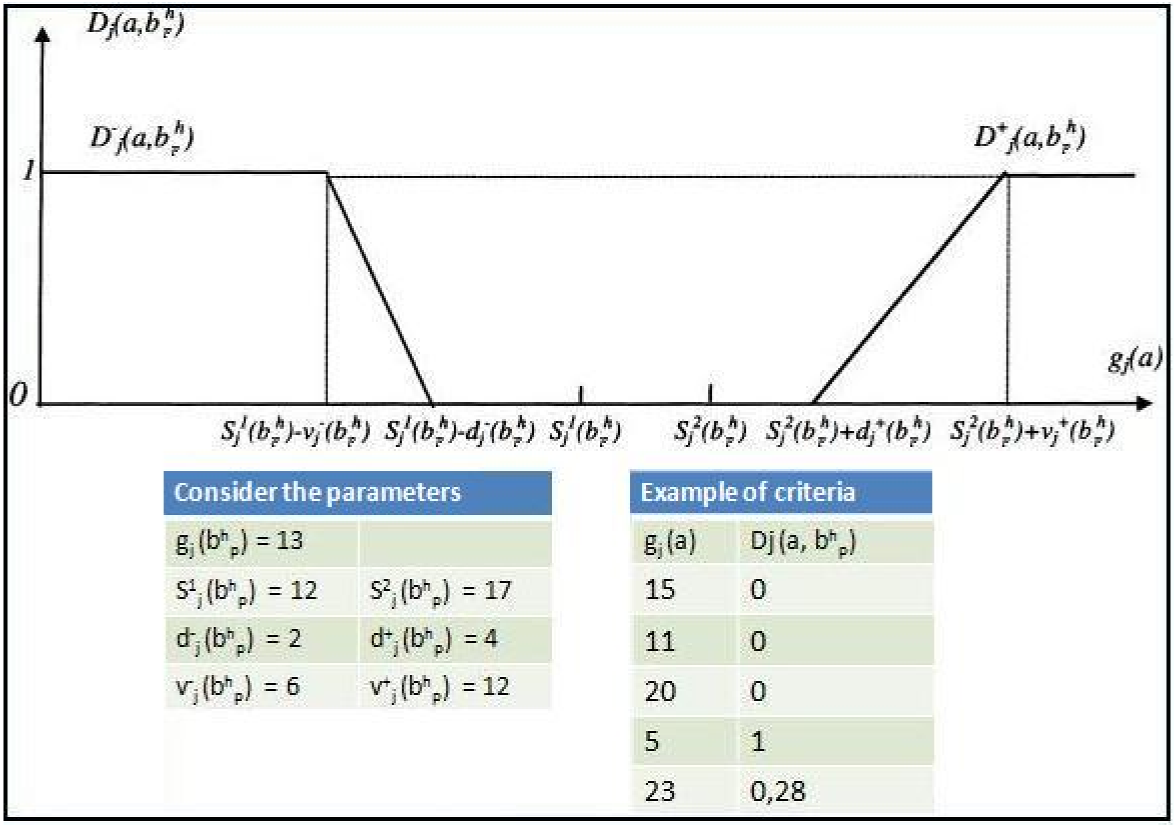
Fig. 2.2: A practical example of C_j computation.

The second step of PROAFTN computes the partial discordance indices. The aim of determining the discordance index $D_j(a, b_p^h)$ of the criterion g_j is to apprehend the fact that such a criterion is more or less discordant with the assertion "a is indifferent to b_p^h" [86]. The discordance index is maximum $(D_j(a, b_p^h) = 1)$ when the criterion g_j uses its veto against this assertion aIb_p^h. It is minimum $(D_j(a, b_p^h) = 0)$ when the criterion is not in discordance with this indifference (i.e. $C_j(a, b_p^h) \notin 0)$.

If the criterion g_j is in discordance (i.e. $C_j(a, b_p^h) = 0$ with indifference and it does not use its veto against this indifference, we have: $0 < D_j(a, b_p^h) < 1$, which represents the intermediary zones between the non-discordance and discordance situations.

The veto thresholds $v_j^-(b_p^h)$ and $v_j^+(b_p^h)$ such as $v_j^+(b_p^h) \geq \mathrm{d}_j^+(b_p^h)$ and $v_j^-(b_p^h) \geq \mathrm{d}_j^-(b_p^h)$, $j = 1, ..., n$, are used to define the values from which the action a is considered as very different to prototype b_p^h for criterion g_j [86].

The discordance index $(D_j(a, b_p^h)$ is represented between the values $S_j^1(b_p^h) - v_j^-(b_p^h)$ and $S_j^1(b_p^h) - d_j^-(b_p^h)$ on one hand and $S_j^2(b_p^h) + v_j^+(b_p^h)$ and $S_j^2(b_p^h) + d_j^+(b_p^h)$ on the other hand, by the linear interpolation function. A didatic example of the calculus of the partial indifference index can be seen in Figure 2.3:



Consider the parameters	
$g_j(b^h_p) = 13$	
$S^1_j(b^h_p) = 12$	$S^2_j(b^h_p) = 17$
$d^-_j(b^h_p) = 2$	$d^+_j(b^h_p) = 4$
$v^-_j(b^h_p) = 6$	$v^+_j(b^h_p) = 12$

Example of criteria	
$g_j(a)$	Dj(a, b^h_p)
15	0
11	0
20	0
5	1
23	0,28

Fig. 2.3: A practical example of D_j computation.

$$D_j(a, b_p^h) = max\{D_j^-(a, b_p^h), D_j^+(a, b_p^h)\}, \tag{2.19}$$

where,

$$D_j^-(a, b_p^h) = \frac{g_j(a) - max\{g_j(a), S_j^1(b_p^h) - d_j^-(b_p^h)\}}{d_j^-(b_p^h) - max\{S_j^1(b_p^h) - g_j(a), v_j^-(b_p^h)\}} \tag{2.20}$$

$$D_j^+(a, b_p^h) = \frac{g_j(a) - min\{g_j(a), S_j^2(b_p^h) + d_j^+(b_p^h)\}}{-d_j^+(b_p^h) + max\{-S_j^2(b_p^h) + g_j(a), v_j^+(b_p^h)\}} \tag{2.21}$$

After that, the next step calculates the fuzzy indifference relation as:

$$I(a,b_i^h) = (\sum_{j=1}^{n} w_j^h C_j(a,b_p^h)) \times (\prod_{j=1}^{n}(1 - D_j(a,b_p^h))^{w_j^h}), \tag{2.22}$$

where w_p^h is a positive coefficient that indicates the importance of an attribute g_j to a class C^h.
The fourth step evaluate the fuzzy membership degree $d(a, C^h)$. The membership degree is computed for each class from the set of categories C by selecting the maximal values of indifference indices from the reference alternatives of each class:

$$d(a,C^h) = max\{I(a,b_1^h), I(a,b_2^h), \ldots, I(a,b_{L_h}^h)\}, h = 1, \ldots, K. \tag{2.23}$$

The final step assigns the alternative to the class with the maximal membership degree:

$$a \in C^h \Leftrightarrow d(a,C^h) = max\{d(a,C^1/l \in \{1,\ldots,K\}\} \tag{2.24}$$

As it can be seen, the implemented methods differ in the way they create relations between the alternatives in order to provide the classification. The PROAFTN method is based on the outranking relation while the gMCDA classifier provides a similarity relation with the application of similarity and dissimilarity thresholds. According to [55], these thresholds represent the maximal difference on the criterion value that is still small enough for two alternatives to be considered similar. In addition, it can be noted that the gMCDA classifier presents less control parameters to be adjusted (only some thresholds and criteria weights). On the other hand, the PROAFTN method encloses a veto threshold, which discriminates situations of discordance with the indifference relation.

2.8 Conclusion

In this chapter, some well-known MCDA classification methods have been introduced. This fundamental background is necessary to provide evidence of some of their advantages and drawbacks. In this context, the next chapter closely investigates some issues related to the nominal classification methods, and presents a novel approach to achieve improved results.

Chapter 3

Proposed Approach to Cope with Multicriteria Nominal Classification

This chapter presents a new decision making approach that combines multidisciplinary techniques with two MCDA nominal classifiers centered on the concepts of prototypes. As described in the previous section, these elements are alternatives that serve as class representatives related to a given problem, which can directly influence the performance index of the classifiers. In this context, in a first moment, the impact that the prototype selection task can exert on such classifiers is investigated. After that, to overcome this problem and improve the performance of the classifiers, two techniques from different research areas that select the best prototypes for a specific problem were employed. The first one is a genetic algorithm, which automatically selects the optimal prototypes. The second one is the ELECTRE IV method, which ranks a set of alternatives based on the DM's preference factors. Finally, the experiments we have carried out, involving different datasets, reveal that there are still some gaps in existing MCDA classification methods that can lead to an improvement of the methodology for the classification problems. This chapter is based on earlier research done by Brasil et al. [13].

3.1 MCDA Nominal Classification Problematics

In the present chapter, two MCDA nominal classifiers, which are developed upon the method recently proposed by Goletsis et al. [123] (gMCDA) and the well-known MCDA PROAFTN classification procedure [86], are employed towards the development of an experimental analysis. Among the different MCDA classification approaches, these two methods were chosen so that we could assign a set of alternatives to a group of non-ordered classes based on similarity measures and outranking relations of the alternatives to the prototype samples.

The MCDA Nominal Classification Methods (NCM) chosen to be investigated have been successfully applied to real-world problems [86, 123]. The major difficulty in applying these methods, however, is that in order to produce models that comply with the DM's expectations, a set of control parameters, such as thresholds, weights, coefficients, etc., needs to be properly set in advance, which turns out to be a difficult task. Some authors, like Belacel [86] and Jacquet-Lagréze and J. Siskos [38], have also provided some approaches to counter this sort of drawback, although their solutions seem to be rather specific to the contexts that were investigated and as yet no general recipes are available to be deployed in all methods and circumstances.

As presented in the last chapter, these classifiers make use of the concept of prototypes, that are special alternatives representing the classes of a problem, and have associated some control parameters related to the expert's preference modeling process with themselves. As the NCMs are centered on this concept, their proper choice can be a significant issue to be dealt with.

This chapter provides a deeper investigation of the NCM problematic through an experimental evaluation. In fact, the experiments provide evidence that the appropriate selection of prototypes as well as the calibration of control parameters are key issues in leveraging the performance of the classifier. Because of this, a new MCDA classification approach is proposed to calibrate their control parameters (cut-off thresholds and prototypes).

3.2 The Impact of Prototype Selection

This section presents some experiments developed to investigate the impact that the prototype selection task exerts on the classification performance. In such regard, the gMCDA and PROAFTN methods were implemented, so that this analysis could be carried out for more than a single classifier. The objective of the experiments is to demonstrate, for five different datasets (Wisconsin Breast Cancer, Iris, Zoo, Balance Scale, and Lung Cancer) available at the UCI repository [20], the sensitivity of the classifier to the selection of the prototypes. Brasil et al. [13] provided the basis for this study.

In order to achieve this objective, ten different groups of prototypes were randomly defined for each dataset. After their selection, the respective alternatives were removed from the original dataset so that they would not serve as references against themselves during the classification process. The number of prototypes chosen was 10% of the original dataset. From that number, the prototypes were separated into their classes respecting their original distribution in the dataset.

For the experiments, thirty different sets of parameter values were arbitrated in a crescent way, beginning with small values, so that the sensitivity of the classifier to the prototypes could be demonstrated for more than a single set of thresholds. The results provided here were summarized by selecting four spaced thresholds sets (1st, 10th, 20th, 30th). The complete results with all the 30 executions for each classifier in every dataset can be found in Appendix A.

Table 3.1 conveys the performance of the classifiers for each group of parameters on the 10 groups of prototypes that were randomly selected for the Wisconsin Breast Cancer dataset. This is a real dataset and it consists of a total of 699 instances (alternatives) and 10 features (criteria) including its class value. The purpose of the gMCDA classifier and the PROAFTN method is to classify those alternatives into two different classes (benign or malign).

For this dataset, the gMCDA classifier presented a uniform performance for the first two sets of parameters, which shows that their selection did not affect the final result. However, with the other two sets, the gMCDA classifier was directly impacted by the choice of the prototypes. For example, the third parameter set presented a performance variation of 18.06% from the second to the sixth execution. Another important condition to be observed is the variation produced by the established thresholds.

The PROAFTN method produced more constant results among different sets of parameters in relation to the gMCDA classifier. However, in the prototype context, the classifier presented significant

Tab. 3.1: Performance of the classifiers when applied to the Wisconsin Breast Cancer dataset with random prototypes.

gMCDA.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	93.66%	91.60%	92.39%	92.71%	91.28%	91.92%	90.33%	90.80%	91.44%	93.82%	92.00%	1.09
10	94.13%	93.34%	92.07%	91.28%	90.80%	91.60%	91.60%	93.34%	91.76%	93.66%	92.36%	1.09
20	33.12%	47.22%	31.53%	41.36%	45.25%	29.16%	34.07%	42.79%	32.49%	34.23%	37.12%	6.08
30	31.38%	46.11%	42.00%	28.21%	28.21%	38.82%	35.34%	26.62%	27.10%	28.68%	33.25%	6.59
Mean	66.32%	71.65%	63.48%	69.74%	69.12%	66.28%	69.49%	67.68%	57.62%	70.09%		
Std. Dev.	27.50	21.31	24.96	27.50	27.86	26.21	27.40	27.01	28.64	30.05		
PROAFTN.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	94.00%	91.37%	91.29%	91.08%	92.88%	89.00%	88.20%	88.44%	87.33%	92.16%	90.58%	2.22
10	92.07%	90.04%	87.34%	90.70%	92.69%	88.58%	89.03%	90.34%	85.22%	91.55%	89.76%	2.28
20	92.84%	86.94%	88.79%	89.18%	91.92%	82.75%	89.85%	89.70%	85.41%	90.51%	88.79%	3.03
30	94.58%	91.81%	92.73%	93.16%	92.88%	91.37%	91.71%	86.13%	87.33%	92.57%	91.43%	2.65
Mean	93.13%	91.03%	89.96%	91.69%	93.17%	88.37%	90.48%	89.89%	90.49%	91.79%		
Std. Dev.	1.90	1.83	2.86	1.86	0.89	2.89	2.34	2.14	2.58	2.45		

differences among the executions. It can be seen that its performance, for the third set of parameters, varies by more than 10% if we consider the first and sixth groups of prototypes.

Table 3.2 presents the experimental results that were conducted on the Iris dataset. This dataset consists of 50 samples from each of three species of Iris flowers (setosa, virginica and versicolor). For each sample, four attributes were measured to determine which species the sample belongs to.

For the IRIS dataset, regardless of the thresholds, the gMCDA classifier presented high variations among the groups of prototypes. In a more general way, for each set of parameters there was at least a variation of 15% among two distinct groups of prototypes. The fourth set of parameters shows a difference of 22.66% between the second and tenth groups.

It is interesting to note that the PROAFTN classifier produces a similar behavior in relation to the gMCDA. For all executions the prototype selection seems to be an important issue to be dealt with. The second set of thresholds, for example, shows a considerable performance difference of 18.49%.

The next experiment contains the ZOO dataset. This dataset consists of 101 alternatives and 17 attributes, including its class value. This is an artificial dataset that has its alternatives assigned into 7 different groups of animals.

As can be seen in table 3.3, the gMCDA classifier is completely affected by the choice of the prototypes and parameters. It is easily noticeable that these elements need to be adjusted to leverage the performance of the classifier. Comparing the choice of the prototypes, if we take the third parameter set as reference, the difference between the pairs of prototypes can be as much as 33.66%. The case gets even worse when we compare only the thresholds, for example, the tenth group of prototypes presents a variation of 98.02%. The characteristic of this dataset (low number of alternatives and high number of attributes or criteria) contributes to this variation, and to an overall low performance of this classifier.

Despite presenting higher and much more uniform performances indices relation to the gMCDA, the PROAFTN method also showed a high sensitivity to the choice of prototypes. In general, the classifier presented a high standard variation between the different groups of prototypes. Under the same threshold parameters (fourth) two executions (second and the seventh) presented results of 71.95%

Tab. 3.2: Performance of the classifiers when applied to the Iris dataset with random prototypes.

gMCDA.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	66.66%	62.00%	56.00%	60.66%	70.66%	66.66%	60.66%	68.00%	60.00%	73.33%	64.46%	5.15
10	74.00%	73.33%	68.00%	75.33%	74.66%	83.33%	75.33%	82.66%	68.66%	80.00%	75.53%	4.93
20	82.00%	70.00%	78.00%	80.66%	80.00%	87.33%	84.66%	84.00%	74.66%	88.00%	80.93%	5.32
30	77.33%	64.00%	66.00%	70.66%	74.66%	80.66%	72.66%	75.33%	65.33%	86.66%	73.33%	6.82
Mean	77.53%	67.84%	68.60%	72.60%	76.93%	81.64%	74.97%	79.06%	67.00%	82.6%		
Std. Dev.	6.54	4.16	6.92	6.09	4.53	7.45	7.51	5.88	4.37	5.09		
PROAFTN.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	83.82%	82.22%	75.36%	79.56%	81.15%	76.81%	80.29%	77.53%	67.64%	82.60%	78.70%	4.73
10	86.76%	80.74%	81.16%	89.05%	92.75%	87.68%	83.94%	91.30%	74.26%	85.51%	85.32%	5.53
20	81.62%	80.00%	74.64%	81.75%	73.19%	86.96%	86.13%	78.26%	77.94%	84.78%	80.52%	4.65
30	78.67%	85.18%	71.73%	73.72%	77.53%	78.98%	89.05%	83.33%	70.58%	85.50%	79.43%	6.26
Mean	85.96%	77.51%	75.97%	82.12%	82.44%	86.14%	87.15%	84.93%	77.03%	87.03%		
Std. Dev.	5.32	7.96	7.85	5.48	7.06	5.00	5.53	6.08	6.83	4.33		

and 44%, respectively, which is a difference of 27.95%.

The fourth analysis included the use of the Balance Scale Tip dataset. This dataset contains 625 alternatives with 5 attributes including its class values (tip to the right, tip to the left or balanced).

For this dataset, the selection of prototypes had its impact minimized in relation to the others. As can be seen in Table 3.4, most of the executions present the performance of the classifiers close to their respective mean, but considering the gMCDA classifier in the same group of prototypes, there are still big performance gaps if we vary the thresholds.

Just like the gMCDA classifier, the PROAFTN analysis demonstrated a constant behavior with the changes of the prototypes. In the worst case, the difference of two executions reached the scale of 10%. However, differently from the gMCDA, the PROAFTN performance levels did not vary that much if we consider the executions over the different set of parameters.

The last experiment was carried out on the Lung Cancer dataset. This dataset is composed of 32 instances with 57 different attributes that can be assigned into 3 different classes. Table 3.5 demonstrates the classification results for this dataset.

Just like for the others datasets, the gMCDA presented a high difference between its executions. In some cases, the difference is up to 24%. The biggest gap can be found between the first and the fifth groups of prototypes of the second set of parameters. For this case, the variation is of 31.04%, making clear the need for some sort of selection of these elements. In the PROAFTN context, if we change only the prototypes, many differences of 20% and even more can be found.

We understand that this sort of empirical assessment is interesting as it reveals how robust/sensitive a MCDA classifier could be to the choice of the alternatives (samples) that serve as class representatives for distinct problems. In fact the experiments involving different datasets from the UCI repository reveal that the proper choice of the prototypes and cut-off thresholds can be a determining issue in leveraging the performance of the classifiers. To overcome this sort of drawback, the next section presents an approach to assist the DM during these activities, especially the prototype selection task.

Tab. 3.3: Performance of the classifiers when applied to the Zoo dataset with random prototypes.

gMCDA.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	93.07%	90.10%	91.09%	93.07%	96.03%	93.07%	96.04%	90.10%	94.06%	98.02%	93.47%	2.51
10	00.00%	00.00%	1.00%	00.00%	00.00%	00.00%	1.00%	00.00%	00.00%	00.00%	00.20%	0.40
20	40.60%	40.60%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
30	4.95%	4.95%	4.95%	33.66%	4.95%	2.97%	4.95%	33.66%	2.97%	4.95%	10.30%	11.70
Mean	23.33%	31.32%	22.93%	18.18%	23.21%	36.70%	17.39%	37.02%	31.45%	35.90%		
Std. Dev.	26.90	24.84	26.78	25.40	27.21	23.56	24.99	23.04	25.65	22.74		
PROAFTN.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	72.50%	64.63%	72.72%	74.07%	62.35%	72.00%	62.66%	73.61%	65.21%	61.03%	68.08%	5.32
10	63.75%	64.63%	71.43%	79.01%	55.29%	61.33%	54.67%	63.89%	59.42%	50.64%	62.40%	8.33
20	77.50%	67.07%	68.83%	81.48%	67.06%	64.00%	58.67%	66.67%	75.36%	74.03%	70.06%	6.89
30	70.00%	71.95%	50.64%	56.79%	55.29%	52.00%	44.00%	62.50%	66.66%	49.35%	57.92%	9.45
Mean	67.33%	66.75%	62.73%	64.90%	59.80%	64.27%	58.13%	61.67%	63.86%	58.96%		
Std. Dev.	7.26	7.29	9.16	9.66%	12.59	9.59	10.73	9.72	7.98	12.56		

Tab. 3.4: Performance of the classifiers when applied to the Balance dataset with random prototypes.

gMCDA.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	54.88%	61.44%	60.32%	56.48%	59.52%	58.87%	62.56%	60.96%	60.16%	56.16%	59.14%	2.52
10	59.84%	67.04%	65.28%	59.84%	66.88%	63.20%	66.88%	63.36%	60.48%	59.51%	63.23%	3.16
20	26.56%	27.20%	27.52%	27.83%	26.40%	24.80%	25.75%	29.92%	34.88%	24.64%	27.55%	3.00
30	10.72%	18.24%	14.24%	16.32%	13.12%	17.76%	19.68%	15.52%	13.91%	11.84%	15.14%	2.90
Mean	40.41%	44.33%	43.01%	39.47%	42.87%	39.60%	43.06%	44.05%	44.17%	38.81%		
Std. Dev.	18.08	18.84	19.17	16.79	19.84	18.74	19.22	18.05	16.98	18.17		
PROAFTN.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	65.84%	60.88%	69.20%	60.70%	63.89%	63.53%	66.54%	74.33%	68.67%	69.38%	66.30%	4.25
10	65.84%	62.83%	63.19%	64.60%	63.54%	62.83%	67.43%	72.57%	68.14%	65.30%	65.63%	3.07
20	77.87%	75.75%	77.69%	66.54%	73.09%	77.16%	71.85%	72.38%	70.26%	72.38%	73.50%	3.64
30	59.11%	65.13%	57.87%	65.66%	62.47%	60.35%	59.46%	69.91%	67.43%	64.60%	63.20%	3.98
Mean	63.66%	64.06%	61.46%	62.10%	63.01%	63.31%	64.68%	68.93%	67.39%	64.25%		
Std. Dev.	7.71	4.63	7.36	5.87	4.92	6.53	5.75	5.11	4.70	7.21		

Tab. 3.5: Performance of the classifiers when applied to the Lung Cancer dataset with random prototypes.

gMCDA.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	55.17%	51.72%	55.17%	62.06%	51.72%	48.27%	51.72%	48.27%	37.93%	55.17%	51.72%	6.29
10	27.58%	44.82%	51.72%	55.17%	58.62%	51.72%	55.17%	44.82%	51.72%	37.93%	47.93%	9.40
20	27.59%	44.83%	44.83%	44.83%	48.28%	44.83%	44.83%	41.38%	48.28%	51.72%	44.14%	6.46
30	37.93%	44.82%	55.17%	41.37%	44.82%	41.37%	55.17%	55.17%	44.82%	44.82%	46.55%	6.35
Mean	40.00%	42.53%	47.36%	46.09%	47.24%	45.86%	49.66%	42.30%	43.68%	47.01%		
Std. Dev.	8.23	6.61	6.77	7.97	7.25	5.88	7.39	8.30	7.03	7.49		
PROAFTN.												
Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	31.03%	34.48%	48.27%	51.72%	37.93%	37.93%	41.37%	48.27%	34.48%	48.27%	41.38%	7.27
10	31.03%	37.93%	34.48%	24.14%	37.93%	44.83%	31.03%	31.03%	37.93%	24.14%	33.45%	6.51
20	51.72%	44.82%	31.03%	44.82%	37.93%	37.93%	51.72%	55.17%	31.03%	44.82%	43.10%	8.49
30	48.28%	44.83%	48.28%	41.38%	44.83%	48.28%	37.93%	44.83%	48.28%	41.38%	44.83%	3.63
Mean	35.17%	36.78%	38.97%	37.59%	37.93%	38.97%	39.54%	38.85%	36.09%	37.93%		
Std. Dev.	9.69	8.07	7.58	7.38	7.84	7.37	10.31	9.28	7.51	8.92		

3.3 Methodologies

Different from most studies in the literature, which put their emphasis on the development of new classification models [55, 86], this section has its focus on the integration of two distinct approaches (see Figure 3.1). The first one describes a genetic algorithm, which can be applied to automatically select the optimal set of prototypes and parameters (direct technique). The second one is the ELECTRE IV method that ranks a set of alternatives based on the DM's preference factors (indirect technique). These complementary approaches can be applied jointly with both classifiers to tackle the problem of prototype selection. In what follows, we provide further details of these methodologies. Figure 3.1 demonstrates different ways to employ these techniques with the implemented classifiers.

3.3.1 Genetic Algorithms

Evolutionary computation is the field of research that draws ideas from evolutionary biology to develop search and optimization techniques (known as evolutionary algorithms) for solving complex problems [112]. Most of these techniques are rooted in the neo-Darwinian theory of evolution, which states that a population of individuals is capable of reproducing and subjected to (genetic) variation followed by selection, resulting over time in new populations of individuals who are increasingly more fit to their environment.

First developed by Holland and his colleagues [66], Genetic Algorithms (GA) are search algorithms that, differently from the traditional optimization and search techniques, are inspired by computational models based on natural evolution. Holland's research was developed with a double aim: (i) to improve the understanding of the adaptive processes of natural systems; and (ii) to design artificial systems that encompass properties and mechanisms of natural systems [33].

According to Goldberg [33], to surpass the traditional optimization and search methods performances, GA introduced the following features:

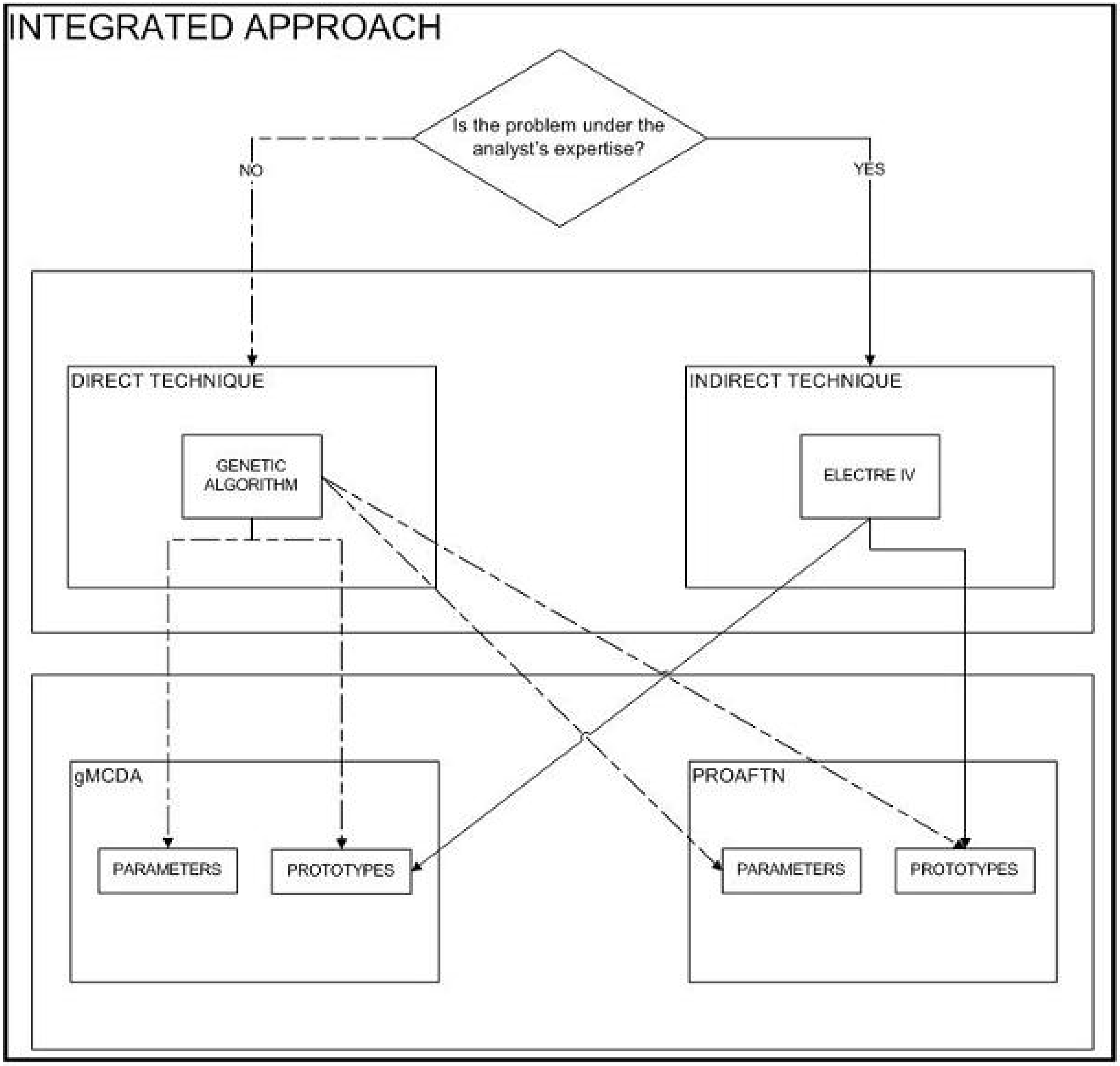


Fig. 3.1: The integrated approach.

- GAs work with an encoding of the parameter set, not the parameters themselves;
- GAs search from a population of points, not a single point;
- GAs use payoff (objective function) information, not derivatives or other auxiliary knowledge;
- GAs use probabilistic transition rules, not deterministic rules.

To enclose the above-mentioned concepts, basically, the GA has to work in the following form:

Basic functioning

As GAs are based on the terms of evolution, it is necessary to clarify some of these terms used throughout this work. The data structures representing the individuals (genotypes) of the population are often called chromosomes; these are one-chromosome (haploid) individuals encoding potential solutions to a problem. In standard GAs, the individuals are represented as strings of bits. Each unit of a chromosome is termed a gene, located in a certain place in the chromosome called the locus. The different values a gene can assume are the alleles. The problem to be solved is captured in an objective (fitness) function that allows evaluating the adequacy of any potential solution.

As each chromosome corresponds to the encoded value of a candidate solution, it has to be decoded into an appropriate form for evaluation and is then assigned a fitness value according to the objective. For each chromosome a probability of reproduction is assigned, so that its likelihood of being selected is proportional to its fitness relative to the other chromosomes in the population. If the fitness of each chromosome is a strictly positive number to be maximized, selection is traditionally performed via an algorithm called fitness proportionate selection [1]. The assigned probabilities of reproduction result in the generation of a population of chromosomes probabilistically selected from the current population. The selected chromosomes will generate offspring via the use of probabilistic genetic operators, namely, crossover (recombination of gene blocks) and mutation (perturbation through genetic variation) each one associated with a specific rate. Each new generation contains a higher proportion of the characteristics of the previous generation's good members, providing a good possibility to converge to an optimal solution of the problem. Figure 3.2 shows the steps of the implemented GA:

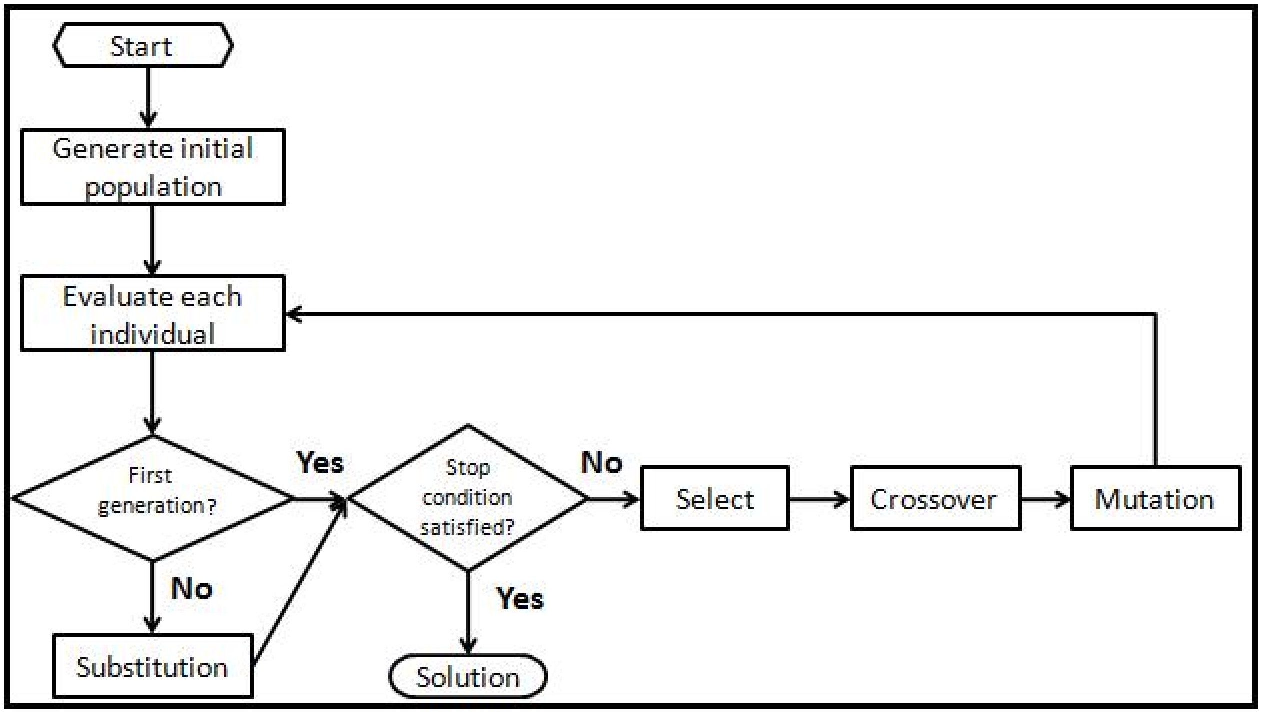


Fig. 3.2: A basic GA.

In order to go through this process and develop a customized GA, it is important to detail its major components. The next subsections present further considerations on them.

Representation

Before developing a customized GA it is necessary to provide the representation of each individual in a correct way. This activity involves the definition of the chromosome design, which needs to be properly done to produce the right representation of the problem being solved [1]. The most widely-used forms of representations are the binary, integer, floating-point, and even more high-level abstractions that could facilitate the genotype-phenotype mapping.

The binary representation was introduced by Holland [66]. Its definition consists of a simple digit that represents the presence (1) or absence (0) of a certain characteristic. Many authors consider it as one of the simplest ways to represent a chromosome [1], and maybe this explains the fact that, historically, many GAs have used this approach to solve their problems.

Although the binary representation is easy to be applied, it is not recommended for all problems. For example, if the problem in consideration maps to a natural situation where the genes can take one specific integer value from a set of values, the most recommend form is the integer representation. On the other hand, if the values we want to represent come from a continuous distribution, it might be necessary to represent the genes as strings of real values.

Fitness Function

When executed, the GA is constantly evaluated under a fitness function, which is the quantification of a specific solution (chromosome). Eiben [1] understands that one of the roles of this evaluation function is to provide the basis for the selection, defining the means of improvement. Furthermore, from another perspective, this function can be seen as the task to be solved in the evolutionary context.

Commonly, the problem under consideration is an optimization problem. In such cases, the literature also uses the name objective function to represent the fitness function. Thus, in some situations the fitness function can be identical to the objective function or a simple transformation of it [1].

Selection

Eiben [1] defines the selection mechanism as a way to distinguish individuals based on their quality. Moreover, this mechanism leverages the presence of better individuals on the next generation, increasing the possibility of obtaining better results.

Before employing a selection mechanism, it is important to evaluate the "exploitation/exploration" balance that is described in [79]. For the author, a strong selection means that highly fit individuals will take over the population, reducing the diversity needed for further change and progress. On the other hand, a too-weak selection will result in too-slow evolution.

The literature presents many distinct techniques to select the individuals, the best known ones are: roulette-wheel, stochastic universal sampling, sigma scaling, elitism, Boltzmann selection, tournament selection, and steady-state selection. This work will focus on the employed techniques, further details about these approaches can be found in [1, 79].

- Fitness-Proportionate Selection (Roulette-Wheel): The Roulette-Wheel method is based on a fitness proportionate selection where the individual's representation is proportional to its respective fitness level. Therefore, considering the fitness of an individual i in the population (f_i), its probability of being selected is:

$$P_i = \frac{f_i}{\sum_{j=1}^{n} f_i}, \tag{3.1}$$

The analogy to a roulette wheel occurs as each potential solution represents a slot on the wheel, where the size of the slots reflect the selection probabilities [1].

- Elitism: Elitism was introduced as an addition to many selection methods trying to prevent the loss of the current fittest member of the population. Such individuals can be lost or replaced if they are not chosen to reproduce or if they are discarded during the process of crossover or mutation. In this context, the GA can retain some number of the best individuals at each generation, which has been demonstrated by many researchers as a mechanism that can increase the GA's performance [1, 79].

Genetic Operators

After the individuals are selected, it is necessary to apply the transformations to the population with the intent of generating new individuals (candidate solutions) from old ones. In general, the purpose of these operators is that they are employed, over the generations, until the search finds a satisfactory result. Mitchell [79] stresses that the choice of the genetic operators is directly related to the encoding strategy.

- Crossover: This genetic operator is responsible for the genetic recombination. This activity is analogous to reproduction and biological crossover, which leads to the diversification of the individuals. In general, the literature highlights three standard forms of recombination: one point crossover, n-point crossover, and the uniform crossover.

  One point crossover was proposed in [66], being the simplest existing recombination form. It works by choosing a random position in the range of [0,l-1], where l is the length of the encoding, and then splitting the parents to form the two children by exchanging the blocks [79].

  A simple generalization of the above-mentioned approach can produce the n-point crossover. For this procedure, two or more crossover points in [0,l-1] are chosen at random, and then the segments are exchanged. In contrast to the one-point crossover, for cases where there is a large number of chromosomes, this approach is less likely to disrupt them. In addition, it can combine more schemas than single-point crossovers [79].

  In contrast to the other two approaches, the uniform crossover works by treating each gene independently and making a random choice as to which parent it should be inherited from [1]. This implementation generates a string of L random variables that vary between 0 and 1. For each position the random value is evaluated on a probability parameter p, and if the value is below it the gene is inherited from the first parent, otherwise from the second.

  Given these different variations of the crossover operator, there is no indication of which one should be always used. In fact, the success or failure of a particular crossover operator is strongly dependent on the particular fitness function, encoding, and other details of the GA [79].

- Mutation: According to Eiben [1], mutation is the name given to the variation operators that use only one parent and create one child through the random individual's structure modification. Furthermore, the author stresses the operator's stochastic characteristic (the mutated child depends on the outcomes of random choices). This operator ensures that the search always has a new direction.

  In order to keep the search evolving, the operator should be employed with a small mutation probability p, otherwise the mutation could lead the GA to a random search. Eiben [1] shows evidence of the dependency of the number of values to be changed to the sequence of random numbers drawn. This way, for an encoding of length l, on average $l \times$ p$valueswillbechanged.$

According to Eiben [1], GAs have successfully been applied to a wide variety of problems, including those which are hard to be solved by other methods. In the MCDA field, their application primarily concerns the task of control parameter optimization [48, 123]. In this work, the control parameter optimization and the prototype selection can be performed by the GA.

3.3.2 ELECTRE IV

According to Zopounidis and Doumpos [23], the indirect techniques are widely used for developing sorting models that employ the outranking concept. To apply this technique, the analyst specifies the parameters based on an interactive interrogation process with the DM. This process ensures that the DM's preferences will be correctly represented.

Despite the significant theoretical work underlying ELECTRE IV, its application is straightforward once the threshold values that define the preference relations are chosen. At this point, the DM's preferences need to be translated into parameters values. For each criterion, one must decide by how much two alternatives need to differ to say that one is weakly or strongly preferred to the other [111].

Unlike other similar algorithms [101], the ELECTRE IV method [17] does not require the analyst to specify a weight for each criterion. This approach avoids the problem of trying to quantify the importance of each criterion for the problem being considered. Conversely, the analyst chooses the criteria that he or she wants to work with and ELECTRE IV combines them to form the outranking relations. Each criterion can be either defined as a benefit or cost criterion. When the analyst considers a cost criterion, the lower the criterion value, the higher its merit and vice versa for a benefit criterion.

Trying to tackle the prototype selection problem and to create a fusion between two different MCDA tasks (ranking and classification) to solve a complex decision problem, we combined these MCDA tasks to try to improve the performance of the classifiers. To employ this method to rank the alternatives of a class, the analyst will have to define only the preference and indifference thresholds for each criterion. In the developed approach, the ELECTRE IV method will respond to the indirect technique for the prototype selection activity.

Basically, the ELECTRE IV method can be divided into seven major stages: 1)Alternatives selection, 2)Criteria selection, 3)Graduation of the alternatives under the selected criteria, 4)Determination of the relative thresholds, 5)Definition of the dominance rules, 6)Construction of the downward and upward ranks, and 7)Determination of the final rank.

The first step to employ the ELECTRE IV algorithm is to select the alternatives that will compose the set A of elements to be ranked. At this stage the decision aiding stakeholders get involved to select the alternatives that will take part in the ranking process.

The second step involves the criteria selection. This activity is commonly performed by the analyst who specifies the important elements that compose an alternative to form outranking relations. In ELECTRE IV, each criterion is treated equally, on the form of same weight when determining outranking relations.

After that, the analyst needs to estimate the selected alternatives under each chosen criterion. This activity requires much effort of the analyst, who needs to infer the criteria values for each alternative.

The fourth stage is the determination of the relative thresholds. This phase basically sets the relation of two alternatives under a given criterion. Based on the fundamental relations proposed by [17] the two alternatives can be defined as: indifferent, strictly preferred, or weakly preferred over a criterion k. They can be stipulated as follows:

- Denoting by x_{ik} and x_{jk} the value of the alternatives i and j on criterion k, they can be set as indifferent if and only if:

  -$q_k \times\ x_{ik} \leq\ x_{ik}$ - $x_{jk} \leq\ q_k \times\ x_{jk}$, for benefit criterion k; and

  -$q_k \times\ x_{jk} \leq\ x_{ik}$ - $x_{jk} \leq\ q_k \times\ x_{ik}$, for cost criterion k.

- Alternative i is strictly preferred to an alternative j over criterion k if and only if:

  x_{ik} - $x_{jk} > p_k \times\ x_{jk}$, for benefit criterion k; and

  x_{ik} - $x_{jk} <$ -$p_k \times\ x_{jk}$, for cost criterion k.

- Alternative i is weakly preferred to an alternative j over criterion k if and only if:

  $q_k \times\ x_{jk} < x_{ik}$ - $x_{jk} \leq\ p_k \times\ x_{jk}$, for benefit criterion k; and

  -$p_k \times\ x_{jk} \leq\ x_{ik}$ - $x_{jk} <$ -$q_k \times\ x_{jk}$, for cost criterion k.

According to Ukkusuri et al. [111], the indifference threshold is employed to account for the imprecision and randomness that affect the input data. On some occasions the analyst faces some difficulties to quantify the thresholds, which can lead the process to unexpected results. An interesting approach to determine this threshold on a specific criterion is to start from a small and non-significant positive value, and gradually increase the value until it gets to a point considered to be the boundary of the difference. In addition, to determine the preference threshold on a criterion, it can be started from a sufficiently large value to ensure a strong preference, and gradually decrease the value down to the limit value so that the strong preference becomes questionable. For the authors, this process can establish the boundary between strong preferences and weak preferences [111].

The fifth stage consists of the definition of the dominance rules. Originally, Roy and Hugonnard [17] established five different dominance rules (quasi-dominance, canonic-dominance, pseudo-dominance, sub-dominance, veto-dominance) to construct the outranking relations for every pair of alternatives. These constructions can vary according to the DM's needs, so that in some problems [108] the authors employ all the constructions, while in others as in [111] the authors can select the group that best suits their needs.

As we want to separate the "good" prototypes from the "bad" ones there is no need to apply all the outranking constructions to provide a detailed rank of all the alternatives. In this context, as employed in [111], two outranking constructions were considered for the prototype selection task implementation. At this point, an alternative i will either strongly or weakly outrank an alternative j based on several restrictions that compare the relative ranks and the thresholds defined in [111]. Below, the outranking constructions applied are detailed.

- Strong Outranking Relation (R_s): It can be said that an alternative i strongly outranks alternative j ($a_i R_s a_j$) if and only if the following conditions are satisfied:

  1) For none of the criteria, a_j is strictly preferred to a_i. 2) The number of criteria on which a_j is weakly preferred to a_i (defined as |J|) does not exceed the number of criteria for which a_i is weakly or strongly preferred to a_j (defined as |K|), in other words, $|K| \geq |J|$.

- Weak Outranking Relation (R_w): On the other hand, to establish a weak outranking relation it is necessary the absence of a strong outranking relation. Moreover, alternative i weakly outranks option j ($a_i R_w a_j$) if and only if at least one of the following two conditions is satisfied.

  1) There is no criterion k such that $x_{jk} > x_{ik} + p_k$, and $|K| < |J|$. 2) There is some criterion k such that x_{ik} + p_k < x_{jk} < $x_{ik} + 2 \times\ p_k$, and a_i is strictly preferred to a_j for at least half of the criteria.

After the matrix with the dominance rules is created, the alternatives need to be ranked through a distillation procedure. This procedure can be divided into two phases: top-down distillation (downward) and bottom-up distillation (upward). The output of this procedure is the ranking of each alternative i from the downward (D_i) and upward (U_i) distillation procedures. Figures 3.2 and 3.3 illustrate the pseudo-code for both approaches [111].

A fundamental step involving both procedures is the calculation of the following items: strengths, weaknesses and qualifications of all alternatives. The strength of the alternative i is defined as the number of alternatives that are strongly out-ranked by alternative i. The weakness of alternative i is the number of alternatives that strongly outrank alternative i. The qualification of an alternative i is computed as its strength subtracted by its weakness.

After the application of the distillation procedures, the last step is to determine the final rank of the alternatives. Here, this rank is set using the mean of the ranks produced by both distillation procedures [2, 111]. This way the final rank is computed as:

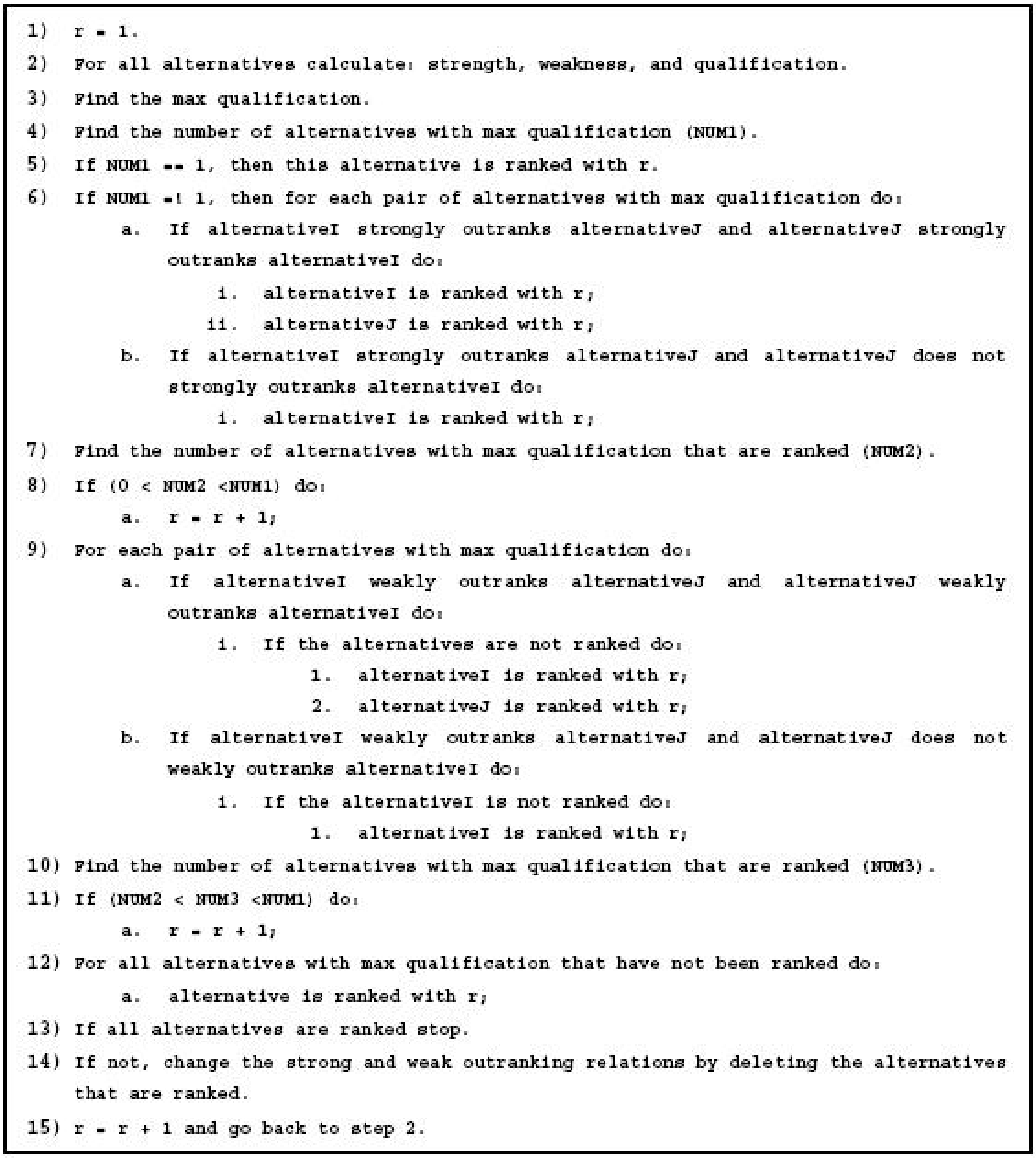
FR_i = 0.5 $\times$ (D_i + U_i), for all i ,

where D_i and U_i are, respectively, the downward and upward rank of an alternative i.

As evidenced in [111], this approach can lead to a situation where more than one alternative can have the same rank.

3.4 Evaluation of the Proposed Approach

In this section, we provide details of the experiments conducted so far over three well-known databases, previously introduced, from the UCI repository [20]. These experiments were also performed to another dataset [9, 12, 11] that will be detailed as a case study in the next chapter. For each database, the procedures were conducted as follows: we have randomly generated 10 pairs of stratified training/test partitions from the original database. After that, for each dataset and execution we have arbitrated one group of thresholds parameters. Finally, to provide the comparison of the methods



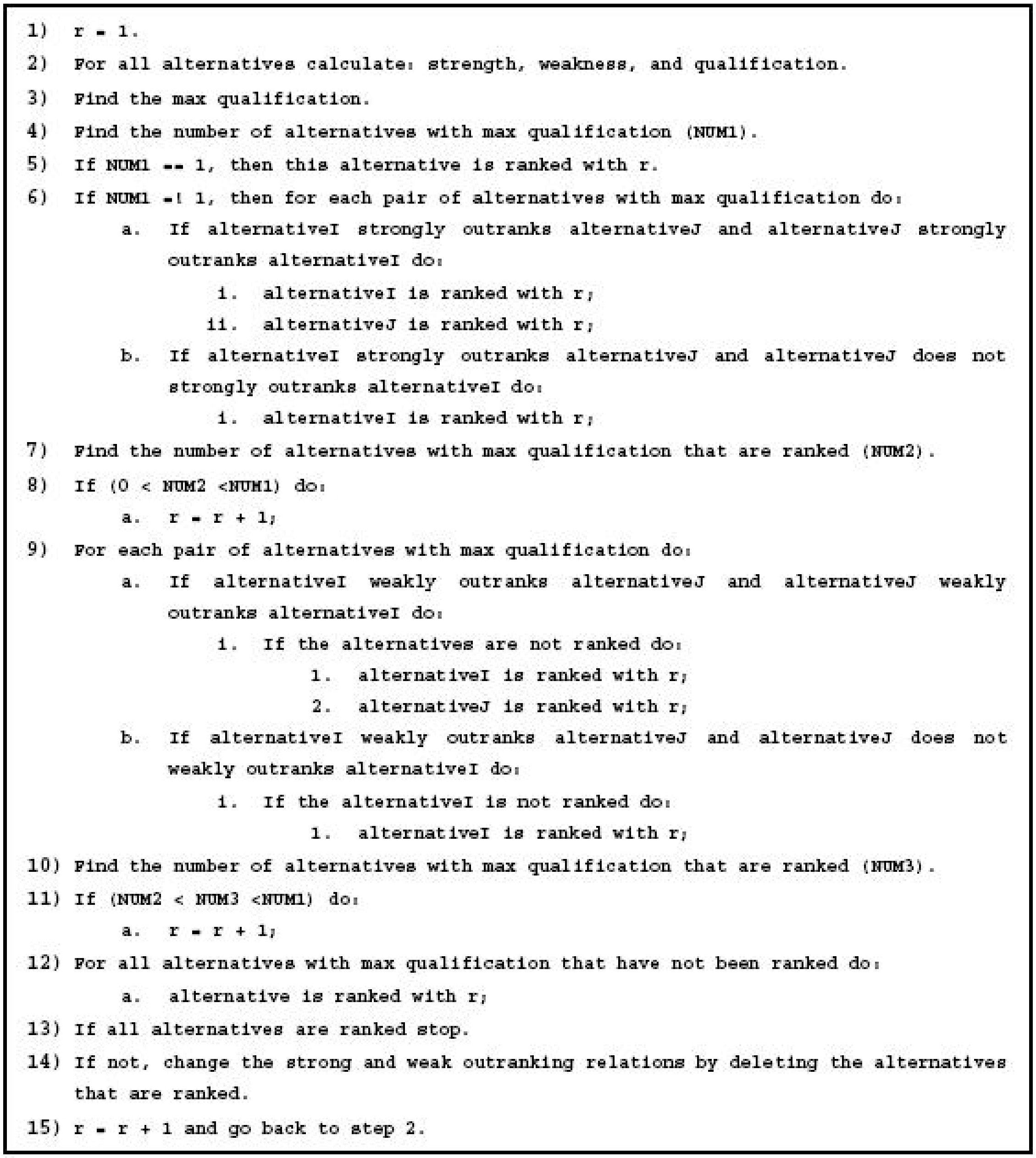
```
1)  r = 1.
2)  For all alternatives calculate: strength, weakness, and qualification.
3)  Find the max qualification.
4)  Find the number of alternatives with max qualification (NUM1).
5)  If NUM1 == 1, then this alternative is ranked with r.
6)  If NUM1 != 1, then for each pair of alternatives with max qualification do:
        a.  If alternativeI strongly outranks alternativeJ and alternativeJ strongly
            outranks alternativeI do:
                i.  alternativeI is ranked with r;
               ii.  alternativeJ is ranked with r;
        b.  If alternativeI strongly outranks alternativeJ and alternativeJ does not
            strongly outranks alternativeI do:
                i.  alternativeI is ranked with r;
7)  Find the number of alternatives with max qualification that are ranked (NUM2).
8)  If (0 < NUM2 <NUM1) do:
        a.  r = r + 1;
9)  For each pair of alternatives with max qualification do:
        a.  If alternativeI weakly outranks alternativeJ and alternativeJ weakly
            outranks alternativeI do:
                i.  If the alternatives are not ranked do:
                        1.  alternativeI is ranked with r;
                        2.  alternativeJ is ranked with r;
        b.  If alternativeI weakly outranks alternativeJ and alternativeJ does not
            weakly outranks alternativeI do:
                i.  If the alternativeI is not ranked do:
                        1.  alternativeI is ranked with r;
10) Find the number of alternatives with max qualification that are ranked (NUM3).
11) If (NUM2 < NUM3 <NUM1) do:
        a.  r = r + 1;
12) For all alternatives with max qualification that have not been ranked do:
        a.  alternative is ranked with r;
13) If all alternatives are ranked stop.
14) If not, change the strong and weak outranking relations by deleting the alternatives
    that are ranked.
15) r = r + 1 and go back to step 2.
```

Fig. 3.3: Downward distillation procedure.

with the random selection of prototypes, for each training/test dataset the following experiments were performed:

- The random selection was executed 10 times, and the mean performance rate achieved by the classifier was calculated;

- The GA engine was executed 10 times, and the mean performance rate achieved by the classifier was calculated;



```
1)  r = 1.
2)  For all alternatives calculate: strength, weakness, and qualification.
3)  Find the max qualification.
4)  Find the number of alternatives with max qualification (NUM1).
5)  If NUM1 == 1, then this alternative is ranked with r.
6)  If NUM1 =! 1, then for each pair of alternatives with max qualification do:
      a.  If alternativeI strongly outranks alternativeJ and alternativeJ strongly
          outranks alternativeI do:
            i.  alternativeI is ranked with r;
           ii.  alternativeJ is ranked with r;
      b.  If alternativeI strongly outranks alternativeJ and alternativeJ does not
          strongly outranks alternativeI do:
            i.  alternativeI is ranked with r;
7)  Find the number of alternatives with max qualification that are ranked (NUM2).
8)  If (0 < NUM2 <NUM1) do:
      a.  r = r + 1;
9)  For each pair of alternatives with max qualification do:
      a.  If alternativeI weakly outranks alternativeJ and alternativeJ weakly
          outranks alternativeI do:
            i.  If the alternatives are not ranked do:
                  1.  alternativeI is ranked with r;
                  2.  alternativeJ is ranked with r;
      b.  If alternativeI weakly outranks alternativeJ and alternativeJ does not
          weakly outranks alternativeI do:
            i.  If the alternativeI is not ranked do:
                  1.  alternativeI is ranked with r;
10) Find the number of alternatives with max qualification that are ranked (NUM3).
11) If (NUM2 < NUM3 <NUM1) do:
      a.  r = r + 1;
12) For all alternatives with max qualification that have not been ranked do:
      a.  alternative is ranked with r;
13) If all alternatives are ranked stop.
14) If not, change the strong and weak outranking relations by deleting the alternatives
    that are ranked.
15) r = r + 1 and go back to step 2.
```

Fig. 3.4: Upward distillation procedure.

- The ELECTRE IV method was employed to rank the prototypes and its performance is calculated.

3.4.1 The Genetic Algorithm Engine

The GA was applied to solve the prototype selection problem through a direct perspective. From this point of view, the algorithm will automatically select the prototypes for each group of the classifier's thresholds parameters. The objective of this algorithm is to optimize the fitness function which is in this case the classifier's classification rate.

To perform the GA experiments, we interpret the chromosome as the prototypes that would be selected. As in all the implemented experiments, the chromosome's size was directly associated to the dataset size, 10% of the original data. To keep the results comparable, it was necessary to maintain in each chromosome the dataset's original distribution of classes. For example, if the Breast Cancer dataset had 65% of benign alternatives and 35% of malign ones, the chromosome would keep that distribution between its genes. The integer codification was applied for each gene, which contained, in the original dataset, the number of the chosen prototype. The initialization phase of the GA implemented, set an initial population of 50 individuals.

The reproductive phase of the algorithm selected the individuals through the Roulette Wheel method, favoring the fitter individuals. Having selected two parents, their chromosomes were recombined applying a crossover, through the single-point approach, and mutation probability of 80% and 15%, respectively. Finally, trying to leverage the GA's performance, maintaining the current fittest member, the elitism technique was employed.

To apply the GA, we separated each original dataset into a training data, which consists of 80% of the dataset, and a test dataset, with the remaining data. For all datasets, the GA stop condition was considered as the number of 500 generations of GA training execution. After that, the best chromosome was applied to the test data. The analysis of the experiments enclosing the GA technique is performed later in this chapter.

3.4.2 The ELECTRE IV Engine

In contrast to the GA, the ELECTRE IV method is applied to solve the prototype selection problem through an indirect perspective. In such case, the analyst is responsible for inputting the preferential parameters (preference and indifference thresholds) of the ELECTRE IV, so that the method can rank the alternatives and the "best" prototypes can be chosen.

Once the alternatives are ranked, our approach selects the same number of prototypes that were assigned in the last experiment, 10% of the original data (maintaining the class distribution). It is interesting to note that the application of the ELECTRE IV method can vary depending on the type of dataset that is under consideration. The datasets chosen present different characteristics, so that we could evidence the intrinsic aspects related to the prototype selection when applying the ELECTRE IV method.

To employ the ELECTRE IV over the Wisconsin Breast Cancer dataset, we established the same preference and indifference thresholds for all criteria ($p = 1.9$ and $q = 0.9$), since they are all benefit criteria and have the same numerical range value. For this dataset we considered the relevance of all criteria, so we avoid discarding any attribute.

The ELECTRE IV application over the IRIS dataset took into consideration a few more aspects than the one made to the Breast Cancer. As it was described in Section 3.2, the IRIS dataset presents three categories (setosa, virginica and versicolor) to which an alternative can be assigned. Making an initial analysis for this problem, we decided to take into consideration, for the classification task, only two out of the four criteria (petal length and petal width). These attributes present a very particular characteristic that easily describes the dataset. The lower values better describe the iris setosa alternatives, the intermediate values describe the iris versicolor, and as these values increases, the probability of the alternative to be an iris virginica turns to be high.

So, for this problem the ELECTRE IV was employed in a very particular way for a three class problem. As described in the ELECTRE IV section, this method can rank an alternative according to a benefit or cost criterion. But in cases where the problem presents more than two classes, the ELECTRE IV should be applied for each class separately, ranking the best alternatives of each class. This occurs because the problems often present conflicting criteria.

Selecting only two criteria (3 and 4), we could define thresholds ($Pc3 = 0.5$, $Pc4 = 5$, $Qc3 = 1$, $Qc4 = 2$), which ranked the alternatives as iris virginica when those criteria presented high values, iris versicolor when these values were intermediate and iris setosa when they were low.

In contrast to the IRIS data, it is a lot harder to employ the ELECTRE IV algorithm when the problem presents an elevated number of criteria, which is the case for the ZOO dataset. This dataset consists of seven different classes, and the criteria are conflicting. This characteristic did not allow us to execute the ELECTRE IV just once to rank all the alternatives. The data was separated, and the alternatives were ranked according to each class.

3.4.3 Results

The results presented here show how effective the employment of the above-mentioned engines could be when dealing with the prototype selection problem. It is important to notice that it was not our purpose to compare the performance of the classifiers in this experiment, as the random choice of thresholds parameters could bring a positive or negative impact to their final performance. The research of Brasil et al. [10] provides another flavor of comparison of gMCDA and PROAFTN classifiers considering only their final performances. The objective here is to compare the random, AG and ELECTRE IV performances with the same set of parameters, prototypes and classifiers. This way, the best prototype selection technique can be verified for both classification methods.

Table 3.6 presents the results of the gMCDA and PROAFTN classifiers over the Iris dataset. As the results provided different meanings, for this dataset, the analysis can be separated by classifiers. When dealing with the PROAFTN classifier, the approaches presented sufficiently good results. The direct technique (GA) increased the classifier performance in 90% of the executions, leading the classifier's performance to significant gains. The indirect technique (ELECTRE IV) demonstrated a good potential to assist the classifier performance on the prototype selection activity. For this method, the seventh execution achieved a classification rate of 100%, being the best among all executions over the Iris dataset. Besides that, the mean performance rate achieved with ELECTRE IV also surpassed the random one, just losing to the GA.

For the gMCDA classifier, the approaches presented distinct results. As in PROAFTN, the GA could increase the classifier performance in 9 out of 10 executions with the mean accuracy rate almost 4% superior, when compared with the random selection. On the other hand, the ELECTRE IV did not perform well over the Iris dataset with the gMCDA classifier. In this case, the ELECTRE IV worsened the performance for all executions, which demonstrate that a good prototype for one classifier is not necessarily good for another.

On Zoo database, Table 3.7, the PROAFTN classifier demonstrated the potential that the prototype selection task can exert over its performance. In a general way, for this dataset, both approaches significantly leveraged the performance of the classifier for all executions. They also presented executions with an accuracy rate of 100% and better best and worst executions than the random selection. In addition, the gMCDA classifier also had its performance increased by both approaches. In contrast

Tab. 3.6: The approach applied over the Iris test data

	PROAFTN			gMCDA		
Execution	Random	GA	Electre IV	Random	GA	Electre IV
1	90.33%	94.33%	96.67%	68.00%	67.67%	30.00%
2	91.67%	95.67%	93.33%	67.00%	69.00%	40.00%
3	90.67%	92.00%	90.00%	64.67%	68.00%	40.00%
4	85.33%	88.67%	86.67%	60.67%	63.33%	30.00%
5	89.67%	91.00%	76.67%	61.00%	70.33%	36.67%
6	91.00%	97.67%	96.67%	70.67%	76.33%	60.00%
7	93.33%	97.00%	100.00%	65.00%	73.67%	43.33%
8	91.00%	94.67%	90.00%	61.33%	71.33%	43.33%
9	91.00%	90.33%	96.67%	64.67%	69.33%	53.33%
10	92.33%	97.00%	96.67%	63.00%	75.33%	33.33%
Mean	90.63%	93.83%	92.33%	64.60%	70.43%	41.00%
Std. Dev	2.13	3.15	6.86	3.27	3.91	9.69
Best	93.33%	97.67%	100.00%	70.67%	76.33%	60.00%
Worst	85.33%	88.67%	76.67%	60.67%	63.33%	30.00%

to the previous dataset, where the ELECTRE IV worsened the gMCDA performance, the indirect technique presented the best performances with excellent classification rates, being 100% correct three times in the classification task.

Finally, the results produced over the Breast Cancer dataset are exhibited in Table 3.8. A general analysis indicates that the GA presented the best classification rates for both classifiers and in all executions, making it the best prototype selection technique for this dataset. Besides the fact that the ELECTRE IV could not leverage the PROAFTN performance for almost all executions, its employment could bring higher results for the gMCDA. For this classification method, the indirect technique increased its performance in almost 30% in relation to the random mean.

The ELECTRE IV method demonstrated potential to solve the prototype selection task. After the analyst sets the preferential information for a problem, the method chooses the prototypes independently of the classifier's parameters. After they are chosen they can be applied to any parameter combination. Another relevant factor that concerns this algorithm is the fact that the analyst and the DM can, depending on the problem, reduce the number of criteria that are used in the prototype selection and the classification purposes. An example of this situation was seen in the IRIS dataset experiment. It is interesting to employ this method for situations where the DM has a vast knowledge of the domain that is being taken into consideration.

While the ELECTRE IV can be applied independently of the parameter set, the GA has to be calibrated according to each classifier's set of thresholds. It is interesting to employ this sort of method when the DM does not have sufficient knowledge of the problem or could not infer the parameters correctly. The GA demonstrated great results over the studied datasets, beating the random selection in all datasets and the ELECTRE IV in a large number of executions. In this context, an overall analysis indicates that the proposed approaches can be a determining issue in leveraging the

Tab. 3.7: The approach applied over the Zoo test data

	PROAFTN			gMCDA		
Execution	Random	GA	Electre IV	Random	GA	Electre IV
1	75.24%	95.71%	95.24%	99.05%	100.00%	100.00%
2	80.48%	90.48%	95.24%	97.14%	93.81%	100.00%
3	78.57%	89.05%	90.48%	96.19%	94.76%	95.24%
4	80.48%	90.95%	100.00%	98.57%	100.00%	100.00%
5	75.71%	90.95%	85.71%	94.76%	95.71%	95.24%
6	66.67%	76.19%	80.95%	89.52%	93.33%	95.24%
7	76.19%	100.00%	90.48%	100.00%	99.52%	100.00%
8	73.81%	92.38%	85.71%	90.48%	91.90%	90.48%
9	69.05%	85.71%	85.71%	89.52%	95.24%	90.48%
10	66.67%	80.95%	76.19%	90.48%	90.00%	95.24%
Mean	74.29%	89.24%	88.57%	94.57%	95.43%	96.19%
Std. Dev	5.22	6.88	7.17	4.21	3.46	3.76
Best	80.48%	100.00%	100.00%	100.00%	100.00%	100.00%
Worst	66.67%	76.19%	76.19%	89.52%	90.00%	90.48%

Tab. 3.8: The approach applied over the Wisconsin Breast Cancer test data

	PROAFTN			gMCDA		
Execution	Random	GA	Electre IV	Random	GA	Electre IV
1	94.14%	95.43%	89.29%	61.86%	92.57%	89.29%
2	95.43%	96.43%	93.57%	64.36%	95.79%	94.29%
3	95.36%	97.64%	92.86%	63.64%	96.43%	95.00%
4	93.14%	95.57%	88.57%	62.86%	94.07%	93.57%
5	93.71%	94.86%	88.57%	60.64%	93.71%	91.43%
6	93.50%	95.14%	90.71%	59.43%	93.00%	87.86%
7	93.86%	95.86%	90.71%	61.64%	93.14%	90.71%
8	93.79%	95.64%	90.00%	63.29%	93.79%	91.43%
9	95.86%	96.64%	92.86%	63.36%	94.64%	92.14%
10	94.07%	95.79%	94.29%	65.14%	94.79%	93.57%
Mean	94.29%	95.90%	91.14%	62.62%	94.19%	91.93%
Std. Dev	0.92	0.81	2.11	1.74	1.23	2.26
Best	95.86%	97.64%	94.29%	65.14%	96.43%	95.00%
Worst	93.14%	94.86%	88.57%	59.43%	92.57%	87.86%

classifier's performance.

As established in the proposed approach (see Figure 3.1), the methodologies can be conjointly applied to provide an integrated classification solution. The next chapter presents some experiments related to the prototype selection activity, but beyond that, it also brings the application of both techniques together, and compares the produced results with some state-of-art classifiers.

3.5 Conclusion

This chapter reveals that the prototype selection is another element of the NCM that can directly influence its performance. The indirect technique was represented by the ELECTRE IV method that needs a DM to select and input its preferential information in order to rank and select the prototypes. It was evidenced that its application improved, in most cases, the performance of the classifiers for many executions and datasets. Its application is indicated in cases where the DM is an expert on that specific domain. The second approach applied to solve the prototype selection problem was the direct or automated one. This technique was implemented through a GA that selected the best prototypes for a problem. Moreover, this approach demonstrated that it could, significantly, improve the classifiers' rate. The next chapter closely investigates the application of these approaches using a case study on Alzheimer's disease.

Chapter 4

Alzheimer's Disease Case Study

This chapter encloses a case study developed upon the gMCDA and PROAFTN classification methods to assist in the early Alzheimer's disease diagnosis process. In order to leverage the performance of the classifiers, the MCDA classification approach presented in Chapter 3 was applied. Various experiments were performed over a novel dataset that takes as reference both the functional and cognitive recommendations and a neuropsychological battery of exams. This chapter is based on the research of Brasil et al. [9, 12].

4.1 Introduction

The Alzheimer's Disease (AD) is a progressive and degenerative disease of the brain which causes a serious impairment over its two main activities: thinking and memory. According to Celsis [93], AD is the most common form of dementia among the elderly population, comprising up to 75% of all dementia cases. AD causes a gradual loss of intellectual abilities with deterioration in cognition, function, and behavior, affecting many aspects of an individual life. Figure 4.1 shows the difference from a normal and an AD brain.

This way, with the decline of the normal functioning over the nervous and other bodily systems, and with the natural behavioral and personality changes, the identification of what constitutes abnormal impairment becomes a hard task. Davidoff [5] argues that the problem over the AD diagnosis is not only related to the current level of understanding of the disease, but also to the comprehension of the normal process involving the elderly. For the author, there are yet no consistent established set of values for what would be a normal level of impairment in the elderly. To overcome these difficulties, some researchers [25, 45, 84] have demonstrated that the AD first symptoms appear relatively early in life, and they evolve during lifetime. This fact raises the chances of identifying the pathology decades before a clinical diagnosis of dementia can be made.

Trying to detect potential patients with AD as early as possible, many studies [7, 8, 64, 99] have investigated some tests and exams that, through a functional and cognitive analysis, may help the early AD detection. In this context, to evaluate the effectiveness of our MCDA classification approach in the early AD detection, we have developed a special-purpose AD-related database by following the recommendations of the Scientific Department of Cognitive Neurology and Aging of the Brazilian Academy of Neurology [103] and by making use of a neuropsychological battery of exams made

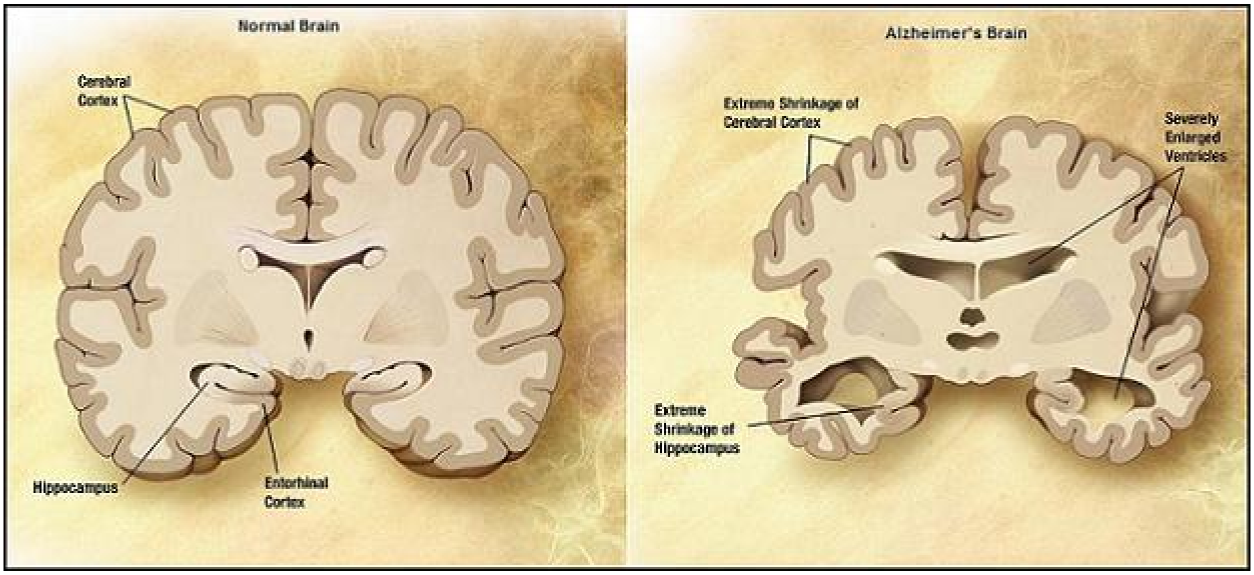

Normal Brain
Cerebral Cortex
Hippocampus
Entorhinal Cortex
Alzheimer's Brain
Extreme Shrinkage of Cerebral Cortex
Severely Enlarged Ventricles
Extreme Shrinkage of Hippocampus

Fig. 4.1: A comparison of a normal and Alzheimer's disease brain. [69]

available by the Consortium to Establish a Registry for Alzheimer's Disease (CERAD) [42]. Various experiments have been performed over this database in a manner as to either fine-tune the components of the MCDA approach or to compare its performance level with that exhibited by other state-of-the-art classification algorithms.

The rest of the chapter is organized as follows. The next section presents an overview of some related work concerning the themes of AD and MCDA classification. The third section provide details of the designed database while the fourth and fifth sections are dedicated to discuss the AD classification experiments conducted over the new database. Finally, the last section concludes the chapter.

4.2 Alzheimer Disease Classification

The very early detection of AD has been deeply investigated in numerous studies in the past years. These studies have demonstrated that the pathology usually arises decades before the clinical diagnosis is effectively made, and so a reliable identification of AD in its earliest stages is one of the major challenges clinicians and researchers face nowadays. The following researches evolved to tackle the identification of AD and other dementia.

Figueiredo et al. [107] present an algorithm that classifies individuals into four different groups (i.e., clinically diagnosed groups of elderly normal, demented, AD, and vascular dementia subjects). The classification is performed after the analysis of computer tomography image data from brain and using an optimal interpolative neural network.

Another classification work related to dementia disorders among the elderly [83] uses a naive credal classifier to address two different classification problems: discrimination between demented and control patients, and the assignment among the different types of dementia. The dataset was developed from a set of measures collected among a series of computerized tests (tasks), which assess some cognitive faculties of the patient.

Sandip et al. [104] realize the AD classification based on a molecular test that evaluates characteristic changes in the concentrations of signaling proteins in the blood, generating a detectable disease-specific molecular phenotype. By this way, through a molecular biomarker in blood plasma, the model classifies the patients into AD or non-AD and identifies those presymptomatic individuals with mild cognitive impairment which will eventually convert to Alzheimer's disease.

In the MCDA field, a decision making model has been recently proposed by Castro and Pinheiro [7, 8] to assist the specialist in the early diagnosis of the Alzheimer's disease. Differently from our approach, this model uses the Macbeth software [24] to construct the judgment matrices and the value scales for each fundamental point of view already defined. Each patient's information is judged by the decision maker for each FPV; then the Macbeth software generates the value scales that will be used in the final judgment of the patient's diagnosis. Instead of providing the classification itself, this sort of model gives the possibilities of a patient acquiring or not a certain type of dementia in the future.

In the present case study, some experiments were performed enclosing the MCDA classification techniques described in Chapter 3 towards the effective early diagnosis of the Alzheimer's disease. As some of the experiments reported here reveal, the appropriate selection of prototypes as well as the calibration of control parameters are key issues to leverage the classifiers' performance.

4.3 Diagnosis of Alzheimer's Disease

Despite the fact that the final AD diagnosis is performed by a microscopic brain tissue exam, through a biopsy or necropsy, Chaves [85] presents evidence that it is not necessary to wait for the patient's death to know, with certainty, what is causing the symptoms and the perceived behavior.

In this context, the doctors can make a "probable" or "possible" AD diagnostic. Pinholt et al. [40] attests the difficult associated with the process of AD detection, wherein the authors highlight that, despite the high incidence of dementia in the elderly population, doctors fail to detect them in 21 to 72% of the cases. There is a vast number of clinical instruments that assist the clinician to obtain the diagnosis. Castro's work [6] summarizes some instruments as follows:

- Complete patient's and family's medical history;
- Neurological exam and neuropsychological tests;
- Review of the use of drugs;
- Physical examination;
- Psychiatric evaluation;
- Laboratorial tests like blood and urine exams;
- Image exams like: computed tomography, magnetic resonance, single photon emission computed tomography (SPECT) and positron emission tomography (PET).

Although these exams are not specific for the AD detection, they can increase its diagnosis accuracy by showing some issues related to its cognitive activity. As a neurodegenerative disease (once the brain cells are lost, they cannot be replaced), the current research challenge is to make a premature diagnose, before the cognitive functioning is impaired [70]. In this context, Rentz e Weintraub [34] stresses that the neuropsychological deficits are still the best way to detect the AD early symptoms.

It is a common sense that the AD early diagnosis can bring benefits to the patients and their families. With the constant development of drug therapies and therapeutic advances, an early treatment can delay the progress of functional decline. The families and all the stakeholders can feel the benefits as they prepare for the patient management, raising their overall quality of life [6].

Besides the quality of life issue, another major factor that stimulates the early diagnosis is the financial one. According to some studies of the Alzheimer's Association, the Alzheimer disease generates high costs, being one of the most expensive diseases, losing only to the cancer and cardiovasculars diseases [102]. The main reason that raises this cost is the need of a multidisciplinary treatment. Normally, the AD affects one or more cognitive areas, such as: attention, perception, memory, reasoning, sense, imagination, thinking and language. For such cases, it may be necessary treatments with psychologists, neurologists, neuropsychologists, geriatricians, psychiatrists, physiotherapists, occupational therapists, etc. Advanced cases can require the constant presence of the family and/or care providers, that are responsible for assisting the patients as they lose their abilities to interpret and express what is happening with their bodies and minds [6, 75].

As mentioned before, this case study seeks to assist the decision maker (clinician) in the early AD diagnosis. To achieve this objective, we have manually designed a specific dataset of cases taking as reference the neuropsychological battery of CERAD standardized assessments and the Brazilian consensus of cognitive and functional evaluation. These are discussed in the following two subsections.

4.3.1 CERAD

The Consortium to Establish a Registry for Alzheimer's Disease was founded in 1986 after the Health Research Extension Act of 1985 with a specific focus on issues of diagnosis and diagnostic standardization [42]. At that time, besides the increasing interest over the illness, there was no uniform guideline over some issues, like diagnostic criteria, testing methods, and classifications of the disease severity, that could be followed. CERAD is a distinctive collaborative initiative that was created to attend to this need. As a prerequisite to the development of a national registry, a high number of universities and research centers joined their efforts to compose the CERAD. The initial objectives of CERAD were: (i) the standardization of the clinic data, neuropsychological, neuropathological and AD's neuroimages evaluation; (ii) features identification and the analysis of the AD natural progress [6].

In this context, CERAD has developed some standardized assessment instruments from different manifestations of Alzheimer's disease: clinical neuropsychology; neuropathology; behavior rating scale for dementia; family history interviews; and assessment of service needs. Of this form, the CERAD battery improved the ability of specialists and researchers to describe and correlate clinical, neuropsychological, and neuropathologic aspects of AD.

4.3.2 The novel dataset

The studies that surround the AD theme have had a substantial increase during the last decades, thus, many advances towards the knowledge over its pathology were made. On the other hand, there are still some issues that were not elucidated yet. Trying to standardize these studies and to find more general guidelines, many countries have published their consensus [26, 44, 47, 103, 115].

In order to provide a way to detect the presence of AD as soon as possible, we have followed the recommendations of the Scientific Department of Cognitive Neurology and Aging of the Brazilian Academy of Neurology [103] while crafting our dataset of cases. This consensus specifies the recommendations over the clinical diagnosis of AD through a functional and cognitive perspective, and therefore the database was designed by following the strategy of correlating clinical and neuropsychological assessments of CERAD with recommendations provided by the Brazilian consensus.

- Language Evaluation: The language is related to our capacity to express or exercise the communication, that can be expressed in two different ways: (i) verbal - when the communication occurs by the expression of the verb through the words and speaking; (ii) non-verbal - when the communication process happens without the verbal presence. This sort of communication is commonly performed through gestures, body language, signs, etc.

  During it first stages, the AD produces semantic-lexical and semantic-discursive problems which may cause the vocabulary forgetfulness and some sort of difficulties in understanding and interpreting the general content of a conversation. As the AD disease evolves over time, there is an increased loss of these functions and an initial modification of the phonological and syntactic signs [18].

  In particular, the language evaluation exams allow for both a quantitative and qualitative diagnosis, showing the profile of the linguistic disorder [103]. For the Brazilian consensus, the Boston Naming Test is one of the recommended tests that can be applied to break down the language aspects of a patient. This way, the first criterion (attribute) considered in the dataset relates to the amount of right answers given by each patient.

- Mini-Mental State Examination: The Mini-Mental State Examination (MMSE) was developed to distinguish psychiatric patients within a functional organical situation [6]. Its employment can assist the clinicians to track the cognitive impairment, being an important screening tool that can guide the evaluation of potential patients with some sort of dementia.

  According to Nitrini [103], this diagnosis should be established with the MMSE application. To comply with the consensus, we turned this assessment into the second criterion associated with each case. The MMSE is composed by many questions grouped into categories that evaluate specific cognitive functions. For the dataset, this criterion reflects the sum of answers correctly assigned by each patient.

- Verbal Fluency and Semantic Memory: The third AD cognitive criterion designates a set of cognitive skills related to social relationships and that guarantee a proper, responsible, and effective conduct of the patient [103]. Among the tests available in CERAD battery, we have used the Verbal Fluency exam. This test requests the patient to verbalize the highest number of animals as possible during a certain period of time. The criterion is defined by the number of items mentioned in a minute, excluding the repeated ones.

- Impairment of Memory: One of the main characteristics of AD is the impairment of memory. The Brazilian consensus [103] stresses the importance of the memory evaluation and suggests the memorization of lists of words as an exam that can be applied to detect any sort of brain impairment during the early stages of the disease. This exam asks the patient to remember a ten-word list after a short period of time to evaluate the status of the short-term memory. The CERAD assessment applies three lists of ten words, so the database criterion we have devised specifies the overall number of words that were remembered by the patient.

- Constructional Ability: The last criterion introduced relates to the concept of constructional ability. This CERAD assessment provides a non-verbal measure of the patient’s mental health through the manipulation of geometric figures. The test requests the patient to draw each figure in two minutes at most (for each). The final analysis is made for each draw separately, and the final score does not exceeds 11. The criterion denotes the patient’s score.

Although the neuropsychological assessments available in the CERAD battery of exams were applied to more than 5.000 patients, only 119 cases could be effectively used in our experiments. This number was achieved after cross-correlating the neuropsychological and clinical assessments in order to certify whether the patient had effectively developed AD or not. By these means, the resulting dataset encompasses 5 criteria and 119 alternatives (cases).The next two sections provide details of the experiments conducted so far over this database.

After that, we perform another experiment considering prototype selection and control parameter calibration tasks conducted, respectively, by the ELECTRE IV and GA engines. Then, we report on the results we have achieved, presenting a comparison with some state-of-the-art classifiers.

4.4 Evaluation of the Proposed Approach: Prototype Selection Problem

In this section, we provide details of the experiments conducted with the employment of the developed model (direct and indirect techniques). The idea is to compare how the random, ELECTRE IV, and GA techniques behave during the prototype selection task. The gMCDA and PROAFTN methods were applied conjointly with the random, direct and indirect techniques, as already exposed during the last chapter’s experiments.

4.4.1 Experiments

The experiments were conducted as follows: we have randomly generated 10 pairs of stratified training/test datasets from the original database and 10 groups of prototypes for each one of these pairs, allocating 80% of the samples for training and the remaining for test. Furthermore, we arbitrated one group of thresholds parameters, for each execution of the classifier, to be used during all the experiment. Finally, to provide the comparison of the methods with the random selection of prototypes, for each training/test dataset the following experiments were performed:

- The random selection was executed 10 times, and the mean performance rate achieved by the classifier was calculated;

Tab. 4.1: Criteria's preference and indifference thresholds

Criteria	Description	+p	+q	-p	-q
C1	Boston Naming Test	0.9	0.39	-0.9	-0.39
C2	Mini-Mental State Examination	0.9	0.39	-0.9	-0.39
C3	Verbal Fluency	1.1	0.45	-1.1	-0.45
C4	Word List	1.1	0.45	-1.1	-0.45
C5	Constructional Praxis	0.9	0.35	-0.9	-0.35

- The GA engine was executed 10 times, and the mean performance rate achieved by the classifier was calculated;
- The ELECTRE IV method was employed to rank the prototypes.

It is important to notice that it was not our purpose to compare the classifiers upon their performances in this experiment, as the random choice of parameters could bring a positive or negative influence to their final performance. The objective is to compare the random, ELECTRE IV and GA performance indices under the same set of parameters, prototypes and classifiers. This way, the best prototype selection technique can be verified for the MCDA nominal classifiers.

ELECTRE IV Engine

As already mentioned, the ELECTRE IV method [17] has been applied to assist in the prototype selection task through an indirect perspective. In such case, the analyst is responsible for providing the system with his/her preferences, which are effectively captured through the preference and indifference parameters (thresholds) associated with ELECTRE IV, so that the method can rank the alternatives.

Since the alternatives are ranked, the number of prototypes chosen while conducting the experiments was 7% of the original dataset. From that number, the prototypes were separated into their classes following their original distribution.

When employed to our AD dataset, as it presents two categories, the ELECTRE IV engine needs to be applied only once to sort the patients from the most probable of not having Alzheimer to those most probable of manifesting the disease. In our experiments, we have ranked the patients from the non-AD to the AD category. For this purpose, we have established the same preference and indifference thresholds for all criteria, as they are all benefit criteria and have the same numerical ranges. For this dataset, all criteria were considered as relevant, so we have avoided discarding any attribute. Table 4.1 shows the preference and indifference values that were elicited for each criterion by the decision maker (clinician).

The Genetic Algorithm Engine

The GA components [1] have been configured as follows: a population of 50 individuals (which initially is randomly generated) is evolved at each generation; the Roulette Wheel operator is used

Tab. 4.2: Integrated approach applied on the Alzheimer test data

	PROAFTN			gMCDA		
Execution	Random Mean	GA Mean	ELECTRE IV	Random	GA Mean	ELECTRE IV
1	70.83%	76.25%	79.17%	58.82%	65.00%	58.33%
2	74.00%	72.40%	79.17%	56.3%	65.20%	80.00%
3	73.75%	81.67%	79.17%	64.71%	65.42%	66.67%
4	75.42%	76.67%	79.17%	58.82%	58.75%	83.33%
5	72.92%	77.92%	79.17%	55.46%	69.58%	66.67%
6	76.67%	68.75%	79.17%	58.82%	61.25%	83.33%
7	76.25%	76.67%	79.17%	67.86%	69.17%	62.50%
8	77.92%	73.33%	79.17%	67.86%	50.83%	79.17%
9	77.92%	74.58%	79.17%	67.86%	53.75%	79.17%
10	68.57%	72.86%	83.33%	70.83%	63.93%	71.43%
Mean	74.42%	75.11%	79.58%	62.73%	62.29%	73.06%
Std. Dev	3.05	3.55	1.32	5.66	6.21	9.11
Best	77.92%	81.67%	83.33%	70.83%	69.58%	83.33%
Worst	68.57%	68.75%	79.17%	55.46%	50.83%	58.33%

to select individuals to reproduce; individuals are recombined through a single-point crossover and the offspring is mutated according to a uniform distribution over the ranges of the parameters; the crossover and mutation rates are 80% and 15%, respectively; and the stop criterion adopted is to go through 500 generations of evolution.

4.4.2 Classification Results

Table 4.2 shows the generated results. A superficial analysis can extract that the ELECTRE IV method obtained the best mean rate on both classifiers. On gMCDA, the random selection and the GA technique presented a similar performance. Though, for the PROAFTN, ELECTRE IV not only obtained the best mean rate as it was the best methodology for all the 10 test sets. Some ELECTRE IV executions augmented the gMCDA classifier performance up to 15% in relation to the random selection. As it can be seen, the prototype selection task improved the classifiers' performance, and once again corroborated with our assumption that is necessary to have some criterion on its selection task.

4.5 An Integrated Approach to Diagnose the Alzheimer's Disease

The experiments described in this section are concentrated on the control parameter calibration and prototype selection tasks conducted, in an integrated approach, respectively, by the GA and ELECTRE IV. The achieved results and a flavor of comparison with some state-of-the-art classifiers

Tab. 4.3: Performance of the classifiers when applied to the 10 test sets.

	1	2	3	4	5	6	7	8	9	10	Mean
gMCDA	91.66%	84%	83.33%	95.83%	91.66%	95.83%	87.5%	95.83%	91.66%	85.71%	90.28%
Random gMCDA	58.82%	56.3%	64.71%	58.82%	55.46%	58.82%	67.86%	67.86%	67.86%	70.83%	62.73%
PROAFTN	79.17%	76%	83.33%	79.17%	83.33%	83.33%	75%	79.17%	83.33%	67.86%	78.97%
Random PROAFTN	70.83%	74%	73.75%	75.42%	72.92%	76.67%	76.25%	77.92%	77.92%	68.57%	74.42%

are presented. The gMCDA and PROAFTN classification methods were applied in order to provide the results.

4.5.1 Experiments

In contrast to the previous section, the intent of this experiment is to provide a comparison of the gMCDA and PROAFTN performance levels over the same circumstances (prototypes and optimized parameters). In such regard, we have decided to compare the gMCDA and the PROAFTN classifiers assisted with the ELECTRE IV and GA engines with their performances when acting alone.

In order to assess the potentials of the whole MCDA approach, we conducted the experiment in the following way: we have randomly generated 10 pairs of stratified training/test datasets from the original database. After that, the ELECTRE IV method is employed to rank and select the best set of prototypes. With the set of prototypes defined, the GA is applied to calibrate the parameters. Finally, with the selected prototypes and calibrated thresholds, the classification task can be performed with the gMCDA and PROAFTN classifiers. The classification performance represents the mean rate of 10 executions over each test set.

As in this experiment the ELECTRE IV method is also responsible for the prototype selection task, its engine is employed under the same conditions as the last experiment did. However, instead of selecting prototypes, the GA is now responsible to set the cut-off parameters, so while its representation and chromosome were adapted (floating-point representation) from the last experiment to contemplate this new activity, the GA's population, operators and generations remained the same.

Tab. 4.4: Performance measures for the AD diagnosis.

Classification Algorithm	Classification Rate (%)
J48	75.63%
NBTree	84.033%
OneR	82.352%
NaiveBayes	75.63%
gMCDA Classification Model	90.28%
PROAFTN Classification Model	78.97%

4.5.2 Classification Results

Table 4.3 shows the performance levels achieved by the classifiers when they had the parameters and prototypes optimized by the developed methodology. It is easily noticeable that the classifiers show a high sensitivity to the choice of prototypes and cut-off threshold values. As it was demonstrated in the last chapter, where the impact of the prototype selection is evidenced, it can be seen that for the AD dataset, the choice of the prototypes and control parameters seems indeed to be a key issue to be properly dealt with in order to leverage the classifiers' performance.

By contrasting the results without the application of the model with those produced by our approach in Table 4.3, it is possible to observe that, for some sets of prototypes, the proposed model could improve the classifiers' performance by more than 20%, taking the mean results over the 10 sets of random prototypes. Moreover, in some runs, the gMCDA classification rate could increase for as high as 33%.

Differently from the last experiment, the results produced by the gMCDA and PROAFTN classifiers can be compared here. Considering only the results produced by the model, the gMCDA classifier surpasses the PROAFTN in 90% of the executions. If we take a single execution, for example the last one (tenth), the difference comes close to 18%.

Finally, to provide a flavor of comparison with other classification algorithms, we have resorted to some well-known classification models available in the WEKA workbench [56]. Table 4.4 brings the average accuracy levels achieved with each contestant model over the 10 derived datasets. The performance level achieved by the gMCDA classifier was superior to those achieved by the other models. It should be emphasized that for each of the four additional classifiers we performed some preliminary experiments in order to manually calibrate its associated control parameters. However, we can not guarantee that the sets of parameters effectively obtained were in fact the optimal ones at all. From the results discussed above, one can conclude that the ELECTRE IV and GA engines have demonstrated good potential in solving the prototype and parameter selection problems.

4.6 Conclusion

The Alzheimer's Disease is a global health problem that is attracting the attention of the public authorities. During the last decades, the worldwide elderly population presented a continuous growth, increasing the incidence of AD. The high costs to treat this disease and the significant deterioration in the families' quality of life have brought a new challenge to researchers and the medical community: identifying the presence of this illness when its first signs appear. In such regard, we designed a new database that takes as reference the functional and cognitive recommendations of the Scientific Department of Cognitive Neurology and Aging of the Brazilian Academy of Neurology and the CERAD's neuropsychological battery of exams. Unlike other studies over the AD, in this case study, our purpose was to assess the performance achieved by an extended version of two MCDA classification algorithms [86, 123] while coping with the AD early diagnosis. In this context, the employment of the ELECTRE IV algorithm revealed that the prototype selection task really exerts an important role over the MCDA classification process. Along with the ELECTRE IV, a GA engine was deployed to assist in the automatic calibration of the control parameter values (weights and thresholds) associated with both classifiers. In a general way, the devised MCDA approach achieved satisfactory levels of accuracy during the patient classification process over the conducted experiments, leveraging the performance of the classifiers and even comparing favorably against some well-known methods [56].

Chapter 5

Conclusions and Future Work

The MCDA classification field has evolved rapidly during the last decades, mainly, with the development of new methods and theories. Even though there is an increasing research in this area, some gaps still remains left as unsolved problems. In contrast to most MCDA classification research, that put more emphasis on the development of new methods, this work reveals that besides the cut-off thresholds, the prototypes exerts a determinant impact over the classifier's performance depending of its selection. To demonstrate that, two nominal classifiers were implemented, so that some experiments could be realized for more than one classifier and with different datasets.

In order to assist in the parameters and prototypes selection, an approach that incorporates techniques from different research areas was developed. Besides the optimization of the cut-off thresholds, the approach encloses two distinct ways of selecting the appropriate prototypes. The first way is based on an indirect technique, which needs an expert to input its preferential information, known as ELECTRE IV. We demonstrated that its application improved, in most cases, the classifier's performance for many groups of parameters. Moreover, the results indicated that it could be applied in cases where the decision analyst is a specialist over a specific domain. The second technique applied to solve the prototype selection problem was the direct or automated one. This approach was implemented with a GA, which selects the best prototypes for a problem, and it demonstrated that it could, significantly, improve the classifier's rate.

To verify the effectiveness of the proposed approach under the prototype selection task, some experiments were employed over some well-known datasets. The results demonstrated that the approach can be employed with the classifiers to leverage their performance. In addition to those experiments, another dataset was considered separately as a case study concerning the Alzheimer's Disease early detection. For this assessment, the dataset was designed taking as reference the CERAD neuropsychological battery of assessments. This battery was chosen because it complies with the recommendations of the Brazilian Academy of Neurology and has been used by several other studies. Overall, the devised MCDA approach could also achieve satisfactory levels of accuracy while classifying the patients in the conducted experiments, leveraging the performance of both classifiers. The average performance level achieved with the gMCDA classifier compares favorably with those achieved with other well-known classifiers [56].

The future work involves the use of GA for the criteria selection problem, this way the classifier can be tested for problems with a high number of criteria. Another interesting investigation is concerned to the number of prototypes that will be selected for each problem. Besides that, it would be interesting to work with other evolutive methods inside the direct technique box 3.1. This experiment could demonstrate a comparison of these techniques when integrated with a NCM. Moreover, a comparative analysis could be realized considering the different ways that the approach can be applied. For example, an experiment could be produced to compare the classifier's performance with the GA technique for parameter calibration and prototype selection against the application of GA for parameters estimation and the ELECTRE IV method for the prototype selection.

Another interesting evolution of the proposed approach could consider the combinations of results of several MCDA classification methods, for instance, the way as it is done in machine learning with voting procedures. As a large number of classification methods have been developed in the artificial intelligence field, in particular, machine learning, the hybridization of such disciplines could bring new results not achieved yet.

In the Alzheimer's Disease case study, we feel that it is possible to elicit novel criteria through the correlation of the Brazilian consensus with other batteries of assessments, which could lead to the design of new datasets. Likewise, the integration of the whole approach developed here with other related MCDA models, such as the one developed in [7, 8], could be a promising strategy to pursue in order to better cope with the early AD diagnosis.

Bibliography

[1] A. Eiben and J. Smith. *Introduction to Evolutionary Computing*. Springer Verlag, 2003.

[2] A. Goicoechea, D.R Hansen, and L. Duckstein. *Multiobjective Decision Analysis with Engineering and Business Applications*. J. Wiley, 1982.

[3] A. Guitouni and J. Martel. Tentative guidelines to help choosing an appropriate MCDA method. *European Journal of Operational Research*, 109:501 – 521, 1998.

[4] A. Newell, J. Shaw, and H. Simon. Chess-playing programs and the problem of complexity. *IBM Journal of Research and Development*, 2(2):320 – 335, 1958.

[5] A.D. Davidoff. Issues in the clinical diagnosis of Alzheimer's disease. *American Journal of Alzheimer's Disease and Other Dementias*, 1(1):9 – 15, 1986.

[6] A.K. Castro. Um Modelo Híbrido Aplicado ao Diagnóstico da Doença de Alzheimer. Master thesis, Master Program in Applied Informatics, University of Fortaleza, 2008.

[7] A.K. Castro, P.R. Pinheiro, and M.C. Pinheiro. A multicriteria model applied in the early diagnosis of Alzheimer's disease: A Bayesian approach. In *Proceedings of VI International Conference on Operational Research for Development*, pages 9 – 19, 2007.

[8] A.K. Castro, P.R. Pinheiro, and M.C. Pinheiro. A multicriteria model applied in the early diagnosis of Alzheimer's disease. In *Rough Sets and Knowledge Technology*, Lecture Notes in Computer Science, pages 612 – 619, 2008.

[9] A.T. Brasil Filho, P.R. Pinheiro, A.L.V. Coelho, and N.C. Costa. Comparison of two mCDA classification methods over the diagnosis of Alzheimer's disease. In *Fourth International Conference on Rough Set and Knowledge Technology (RSKT 2009)*, Lecture Notes in Computer Science, pages 289 – 298, 2009.

[10] A.T. Brasil Filho, P.R. Pinheiro, A.L.V. Coelho, and N.C. Costa. Comparison of two prototype-Based multicriteria classification methods. In *IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MCDM 2009)*, IEEE Symposium Series on Computational Intelligence 2009, 2009.

[11] A.T. Brasil Filho, P.R. Pinheiro, A.L.V. Coelho, and N.C. Costa. Selecting prototypes for two multicriteria classification methods: A comparative study. In *World Congress on Nature and Biologically Inspired Computing (NABIC 2009)*, 2009.

[12] A.T. Brasil Filho, P.R. Pinheiro, and A.L.V. Coelho. Towards the early diagnosis of Alzheimer's disease via a multicriteria classification model. In *5th Evolutionary Multi-Criterion Optimization (EMO 2009)*, Lecture Notes in Computer Science, pages 393 – 406, 2009.

[13] A.T. Brasil, P.R. Pinheiro, and A.L.V. Coelho. *Innovations and Advances in Computer Sciences and Engineering*, chapter The Impact of the Prototype Selection on a Multicriteria Decision Aid Classification Algorithm (to appear). Springer Verlag, 2009.

[14] B. Roy. The outranking approach and the foundations of ELECTRE methods. *European Journal of Operational Research*, 31:49 – 73, July 1991.

[15] B. Roy. *Multicriteria Methodology for Decision Aiding*. Kluwer Academic Publishers, 1996.

[16] B. Roy. *Multiple Criteria Decision Analysis: State of the Art Surveys Series*, volume 78 of *International Series in Operations Research Management Science*, chapter Paradigms and Challenges, pages 3 – 24. Springer Verlag, 2005.

[17] B. Roy and B. Hugonard. Ranking of suburban line extension projects on the Paris metro system by a multicriteria method. *Transportation Research*, 16:301 – 312, 1982.

[18] B.P. Damasceno. Envelhecimento cerebral: O problema dos limites entre o normal e o patológico. *Arq. Neuropsiquiatria*, 57(1):78 – 83, 1999.

[19] C. Bana e Costa, L. Ensslin, E.C. Correa, and J.C. Vansnick. Decision support systems in action: Integrated application in a multicriteria decision aid process. *European Journal of Operational Research*, 133:315 – 335, 1999.

[20] C. L. Blake and C. J. Merz. UCI Repository of machine learning databases, June 2009. `http://archive.ics.uci.edu/ml/`.

[21] C. Zopounidis and M. Doumpos. Business failure prediction using the UTADIS multicriteria analysis method. *Journal of the Operational Research Society*, 50(11):1138 – 1148, 1999.

[22] C. Zopounidis and M. Doumpos. A preference disaggregation decision support system for financial classification problems. *European Journal of Operational Research*, 130(2):402 – 413, 2001.

[23] C. Zopounidis and M. Doumpos. Multicriteria classification and sorting methods: A literature review. *European Journal of Operational Research*, 138(2):229 – 246, 2002.

[24] C.A. Bana e Costa, J.M. Corte, and J.C. Vasnick. Macbeth. Jornal O Estado de São Paulo, 2003. LSE-OR Working Paper.

[25] C.H. Kawas, M.M. Corrada, R. Brookmeyer, A. Morrison, S.M. Resnick, A.B. Zonderman, and D. Arenberg. Visual memory predicts Alzheimer's disease more than a decade before diagnosis. *Neurology*, 60(7):1089 – 1093, 2003.

[26] C.J.S. Patterson, S. Gauthier, H. Bergman, and others. The recognition, assessment and management of dementing disorders: Conclusions from the Canadian Consensus Conference on Dementia. *Canadian Medical Association Journal*, 160,(2), 1999.

[27] C.V. Altrock. *Fuzzy Logic and Neurofuzzy Applications in Business and Finance*. Prentice Hall, 1996.

[28] D. Dubois and H. prade. *Advances in Fuzzy Set Theory and Applications*, chapter Decision-making under fuzziness, pages 279 – 302. North-Holland, 1979.

[29] D. Lowe. Similarity metric learning for a variable-kernel classifier. *Neural Computing*, 7:72 – 85, 1995.

[30] D. R. Wilson and T. R. Martinez. Reduction techniques for instance based learning algorithms. *Machine Learning*, 38(3):257 – 286, 2000.

[31] D. Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions Systems, Man and Cybernetics*, 2:408 – 421, 1972.

[32] D.C. Morais. Modelagem Multicritério em Grupo para Planejamento Estratégico do Controle de Perdas no Abastecimento de Água. Phd thesis, Programa de Pós-Graduação em Engenharia de Produção, Federal University of Pernambuco, 2006.

[33] D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

[34] D.M. Rentz and S. Weintraub. *Early Diagnosis of Alzheimer's Disease*, chapter Neuropsychological Detection of Early Probable Alzheimer's Disease, pages 169 – 189. Humana Press, 2000.

[35] D.W. Aha, D. Kibler, and M.K. Albert. Instance-based learning algorithms. *Machine Learning*, 6(1):37 – 66, 1991.

[36] E. Georgopoulou, Y. Sarafidis, S. Mirasgedis, S. Zaimi, and D. Lalas. A multiple criteria decision-aid approach in defining national priorities for greenhouse gases emissions reduction in the energy sector. *European Journal of Operational Research*, 146(1):199 – 215, 2003.

[37] E. Jacquet-Lagrèze and Y. Siskos. Assessing a set of additive utility functions for multicriteria decision making: The UTA method. *European Journal of Operational Research*, 10:151 – 164, 1982.

[38] E. Jacquet-Lagrèze and Y. Siskos. Preference disaggregation: 20 years of MCDA experience. *European Journal of Operational Research*, 130(2):233 – 245, 2001.

[39] E. Paschetta and A. Tsoukiàs. A real world MCDA application: Evaluating software. *Journal of Multi-Criteria Decision Analysis*, 9:205 – 226, 2000.

[40] E.M. Pinholt, K. Kroenke, J.F. Hanley, M.J. Kussman, P.L. Twyman, and J.L. Carpenter. Functional assessment of the elderly: A comparison of standard instruments with clinical judgment. *Archives of Internal Medicine*, 147:484 – 488, 1987.

[41] F. Joerin and A. Musy. Land management with GIS and multicriteria analysis. *International Transactions in Operational Research*, 7(1):67 – 78, 2000.

[42] G. Fillenbaum, G. van Belle, J. Morris, R. Mohs, S. Mirra, P. Davis, P. Tariot, J. Silverman, C. Clark, and K. Welsh-Bohmer. Consortium to establish a registry for Alzheimer's disease (CERAD): The first twenty years. *Alzheimer's & Dementia*, 4:96 – 109, 2008.

[43] G. Munda, P. Nijkamp, and P. Rietveld. Qualitative multicriteria methods for fuzzy evaluation problems: An illustration of economic-ecological evaluation. *European Journal of Operational Research*, 82(2):79 – 97, 1995.

[44] G.W. Small, P.V. Rabins, P.P. Barry, and others. Diagnosis and treatment of Alzheimer disease and related disorders: Consensus statement of the American Association for Geriatric Psychiatry, the Alzheimer's Association, and the American Geriatrics Society. *The Journal of the American Medical Association*, 278,(16), 1987.

[45] H. Braak and E. Braak. Frequency of stages of Alzheimer-related lesions in different age categories. *Neurobiology of Aging*, 18:351 – 357, 1997.

[46] H. Brighton and C. Mellish. Advances in instance selection for instance-based learning algorithms. *Data Mining and Knowledge Discovery*, 6(2):153 – 172, 2002.

[47] H. Chertkow. Diagnosis and treatment of dementia: Introducing a series based on the third Canadian Consensus Conference on the diagnosis and treatment of dementia. *Canadian Medical Association Journal*, 178(3):316 – 321, 2008.

[48] H.A. Guvenir and E. Erel. Multicriteria inventory classification using a genetic algorithm. *European Journal of Operations Research*, 105(1):29 – 37, 1998.

[49] H.A. Simon. *Administrative Behavior: A Study of Decision-Making Processes in Administrative Organizations*. Free Press, 4th edition, 1997.

[50] H.G. Costa, A.F.U. Mansur, A.L.P. Freitas, and R.A. Carvalho. ELECTRE TRI aplicado a avaliação da satisfação de consumidores. *Produção*, 17(2):230 – 245, 2007.

[51] H.G. Costa, S.S Motta, and R.H. Gutierrez. Avaliação da produção docente: Abordagem multicritério pelo método ELECTRE II. In *XXVI ENEGEP*, Fortaleza, CE, October 2006.

[52] H.J. Zimmerman. Fuzzy programming and linear programming with several objective functions. *Fuzzy Sets and Systems*, 1:45 – 55, 1978.

[53] I. Ashikhmin and E. Furems. UniComBOS - Intelligent decision support system for multicriteria comparison and choice. *Journal of Multi-criteria Decision Analysis*, 13(2):147 – 157, 2005.

[54] I. Tamanini, P.R. Pinheiro, A.L. Carvalho, and A.K. Castro. A novel multicriteria model applied to cashew chestnut industrialization process. In M. Koppen J. Mehnen, A. Tiwari and Ashraf Saad, editors, *Applications of Soft Computing from Theory to Praxis*, volume 58, 2009.

[55] I. Yevseyeva. Solving Classification Problems with Multicriteria Decision Aiding Approaches. Phd thesis, Faculty of Information Technology, University of Jyväskylä, 2007.

[56] I.H. Witten and E.Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2nd edition, 2005.

[57] I.M. Makarov, T.M. Vinogradskaya, A.A. Rubchinsky, and V.B. Sokolov. *The Theory of Choice and Decision Making*. Mir. Publishers, 1987.

[58] J. Devaud, G. Groussaud, and E. Jacquet-Lagreze. UTADIS: Une méthode de construction de fonctions d'utilité additives rendant compte de jugements globaux. In *Proceedings of European working group on MCDA*, Bochum, Germany, 1980.

[59] J. Figueira, V. Mousseau, and B. Roy. *Multiple Criteria Decision Analysis: State of the Art Surveys Series*, volume 78 of *International Series in Operations Research and Management Science*, chapter Electre methods, pages 133 – 162. Springer Verlag, 2005.

[60] J. Fodor and M. Roubens. *Fuzzy Preference Modelling and Multicriteria Decision Support*. Kluwer Academic Publishers, 1994.

[61] J. Martel and B. Matarazzo. *Multiple Criteria Decision Analysis: State of the Art Surveys Series*, volume 78 of *International Series in Operations Research and Management Science*, chapter Other outranking approaches, pages 197 – 264. Springer Verlag, 2005.

[62] J. Siskos. A way to deal with fuzzy preferences in multicriteria decision problems. *European Journal of Operational Research*, 10:314 – 324, 1982.

[63] J. von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 3rd edition, 1980.

[64] J.A. Mortimer, A.R. Borenstein, K.M. Gosche, and D.A. Snowdon. Very early detection of Alzheimer neuropathology and the role of brain reserve in modifying its clinical expression. *Journal of Geriatric Psychiatry and Neurology*, 18(4):218 – 223, 2005.

[65] J.C.C.B.S. Mello, E.G. Gomes, F.R. Leta, and R.B.V. Pessolani. Conceitos básicos do apoio multicritério à decisão e sua aplicação no projeto aerodesign. *Engevista*, 5(8):22 – 35, 2003.

[66] J.H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

[67] J.P. Brans and B. Mareschal. *Multiple Criteria Decision Analysis: State of the Art Surveys Series*, volume 78 of *International Series in Operations Research and Management Science*, chapter PROMETHEE methods, pages 163 – 196. Springer Verlag, 2005.

[68] J.P. Brans and P. Vincke. A preference ranking organization method. *Management Science*, 31(6):647 – 656, 1985.

[69] D. Kang. Protein Increases Production of Amyloid Beta in Alzheimer's Disease, June 2009. `http://www.elements4health.com/protein-increases-\~production-of-amyloid-beta-in-alzheimers-disease.html`.

[70] K.R. Daffner and L.F.M. Scinto. *Early Diagnosis of Alzheimer's Disease*, chapter Early Diagnosis of Alzheimer's Disease: An Introduction, pages 1 – 27. Humana Press, 2000.

[71] L. Yu, S. Wang, and K.K. Lai. An intelligent-agent-based fuzzy group decision making model for financial multicriteria decision support: The case of credit scoring. *European Journal of Operational Research*, 195(3):942 – 959, 2009.

[72] L. Zadeh. Fuzzy sets. *Information Control*, 8:338 – 353, 1965.

[73] L.F.A.M. Gomes and L.L. Santos. An application of the ELECTRE Tri method to human resource management in telecommunications in Brazil. *Rio's International Journal on Sciences of Industrial and Systems Engineering and Management*, 2:1 – 20, 2008.

[74] L.F.A.M. Gomes, C.F.S. Gomes, and A.T Almeida. *Tomada de Decisão Gerencial:Enfoque Multicritério*. Atlas, 2nd edition, 2006.

[75] L.P. Gwyther. Care of Alzheimer's patients: A manual for nursing home staff. Technical report, American Health Care Association, 1985.

[76] M. Doumpos and C. Zopounidis. The use of the preference disaggregation analysis in the assessment of financial risks. *Fuzzy Economics Review*, 31:39 – 57, 1998.

[77] M. Doumpos and C. Zopounidis. *Multicriteria Decision Aid Classification Methods*. Kluwer Academic Publishers, 2002.

[78] M. Doumpos and C. Zopounidis. Developing sorting models using preference disaggregation analysis: An experimental investigation. *European Journal of Operational Research*, 154(3):585 – 598, 2004.

[79] M. Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, 1998.

[80] M. Oral and O. Kettani. Modelling the process of multiattribute choice. *Journal of the Operational Research Society*, 40(3):281 – 291, 1989.

[81] M. Rogers, M. Bruen, and L. Maystre. *ELECTRE and Decision Support*. Kluwer Academic Publishers, 1st edition, 1999.

[82] M. Schneider, A. Kandel, G. Langholz, and G. Chew. *Fuzzy Expert System Tools*. J. Wiley, 1996.

[83] M. Zaffalon, K. Wesnes, and O. Petrini. Reliable diagnoses of dementia by the naive credal classifier inferred from incomplete cognitive data. *Artificial Intelligence in Medicine*, 29:61 – 79, 2003.

[84] M.F. Elias, A. Beiser, P. Wolf, R. Au, R.F. White, and R.B. D'Agostino. The preclinical phase of Alzheimer disease: A 22-year prospective study of the Framinghan cohort. *Archives of Neurology*, 57:808 – 813, 2000.

[85] M.I.F. Chaves. Diagnóstico Diferencial das Doenças Demenciantes. In O.V. Forlenza and P. Caramelli, editors, *Neuropsiquiatria Geriátrica*, pages 81 – 104. Atheneu, 2000.

[86] N. Belacel. Multicriteria assignment method PROAFTN: Methodology and medical applications. *European Journal of Operational Research*, 125:175 – 183, 2000.

[87] O. Larichev. A study on the internal organization of expert knowledge. *Pattern Recognition and Image Analysis*, 5(1):57 – 63, 1995.

[88] O. Larichev, A. Asanov, and Y. Naryzhny. Effectiveness evaluation of expert classification methods. *European Journal of Operational Research*, 138(2):260 – 273, 2002.

[89] O. Larichev, A. Kortnev, and D. Kochin. Decision support system for classification of a finite set of multicriteria alternatives. *Decision Support Systems*, 33(1):13 – 21, 2002.

[90] O. Larichev, A. Mechitov, V. Morgoev, H. Moshkovich, and E. Furems. *Contemporary Issues in Decision Making*, chapter Exact duplicates of human judgments, pages 43 – 55. Elsevier, 1990.

[91] O. Larichev and H. Moskovich. Decision support system ORCLASS. In *Tenth International Conference on Multiple Criteria Decision Making*, volume 4, pages 341 – 350, Taipei, Taiwan, 1992.

[92] O. Larichev and H. Moskovich. *Verbal Decision Analysis for Unstructured Problems*. Kluwer Academic Publishers, 1997.

[93] P. Celsis. Age-related cognitive decline, mild cognitive impairment or preclinical Alzheimer's disease? *Annals of Medicine*, 32:6 – 14, 2000.

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

[95] P. Hart. The condensed nearest neighbor rule. *IEEE Transactions on Information Theory*, 14:515 – 516, 1968.

[96] P. Korhonen. *Multiple Criteria Decision Analysis: State of the Art Surveys Series*, volume 78 of *International Series in Operations Research and Management Science*, chapter Interactive Methods, pages 641 – 665. Springer, 2005.

[97] P. Vincke. *Multicriteria Decision-Aid*. J. Wiley, 1992.

[98] P. Willet, J.M. Bernard, G.M. Downs. Chemical Similarity Searching. *Journal of Chemical Information and Computer Sciences*, 38:983 – 996, 1998.

[99] P.J. Nestor, P. Scheltens, and J.R. Hodges. Advances in the early detection of Alzheimer's disease. *Neurodegeneration*, 5:S34 – S41, 2004.

[100] R. Bellman and L. Zadeh. Decision-Making in a Fuzzy Environment. *Management Science*, 17, 1970.

[101] R. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press, 2008.

[102] R. Koppel. Alzheimer's disease: The costs to U.S. businesses in 2002. Report, Alzheimers Association, 2002.

[103] R. Nitrini, P. Caramelli, C.M. Bottino, B.P. Damasceno, S. M. Brucki, and R. Anghinah. Diagnóstico de doença de Alzheimer no Brasil: Avaliação cognitiva e funcional, recomendações do Departamento Científico de Neurologia Cognitiva e do Envelhecimento da Academia Brasileira de Neurologia . *Arquivos de Neuro-Psiquiatria*, 63(3 - A):713 – 719, 2005.

[104] R. Sandip and others. Classification and prediction of clinical Alzheimer's diagnosis based on plasma signaling proteins. *Nature Medicine*, 13:1359 – 1362, 2007.

[105] R. Slowinski and C. Zopounidis. Classement et choix en présence de points de vue multiples: La méthode ELECTRE. *Revue Francaise d'Informatique et de Recherche Operationnelle*, 8:57 – 75, 1968.

[106] R. Slowinski and C. Zopounidis. Application of the rough set approach to evaluation of bankruptcy risk. *International Journal of Intelligent Systems in Accounting, Finance and Management*, 4(2):27 – 41, 1995.

[107] R.J.P. Figueiredo, W.R. Shankle, A. Maccato, M.B. Dick, P. Mundkur, I. Mena, and C.W. Cotman. Neural-network-based classification of cognitively normal, demented, Alzheimer disease and vascular dementia from single photon emission with computed tomography image data from brain. *Medical Sciences*, 92(12):5530 – 5534, 1995.

[108] R.M. Moreira. Análise Multicritério dos Projetos do Sebrae/RJ através do ELECTRE IV. Master thesis, Programa de Pós-Graduação e Pesquisa em Administração e Economia, Faculdade de Economia e Finaças IBMEC, 2007.

[109] R.R. Yager. Multiple objective decision-making using fuzzy sets. *International Journal of Man-Machine Studies*, 9:375 – 382, 1977.

[110] S. Greco, B. Matarazzo, and R. Slowinski. *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory, and Applications*, chapter The use of rough sets and fuzzy sets in MCDM, pages 14.1 – 14.59. Kluwer Academic Publishers, 1999.

[111] S.V. Ukkusuri, A. Karoonsoontawong, K.M. Kockelman. *New Transportation Research Progress*, chapter Congestion Pricing Technologies: A Comparative Evaluation in New Transportation Research Progress, pages 121 – 142. Nova Science Publishers, 2007.

[112] T. Back, D. Fogel, and Z. Michalewicz. *Handbook of Evolutionary Algorithms*. Oxford Press, 1997.

[113] T. Saaty. *The Analytic Hierarchy Process, Planning, Priority Setting, Resource Allocation*. McGraw-Hill, 1980.

[114] V. Belton and T. Stewart. *Multiple Criteria Decision Analysis: An Integrated Approach*. Kluwer Academic Publishers, 2002.

[115] V. Bonavita, C. Caltagirone, P.L. Canonico. Malattia di Alzheimer documento di consenso: Società Italiana di Neuroscienze. *Giorn. Gerontol.*, 48:75 – 91, 2000.

[116] V. Mousseau. A general framework for constructive learning preference elicitation in multiple criteria decision aid. Technical report, LAMSADE, Université Paris-Dauphine, 2005.

[117] V. Mousseau and R. Slowinski. Inferring an ELECTRE-TRI model from assignment examples. *Journal of Global Optimization*, 12(2):157 – 174, 1998.

[118] V. Mousseau, J. Figueira, and J. Naux. Using assignment examples to infer weights for ELECTRE TRI method: Some experimental results. *European Journal of Operational Research*, 130(2):263 – 275, 2001.

[119] V. Mousseau, L. Dias, and J. Figueira. On the notion of the category size in multiple criteria sorting models. Technical report, Center for Discrete Mathematics and Theoretical Computer Science, Rutgers, 2003.

[120] V. Mousseau, R. Slowinski, and P. Zielniewicz. ELECTRE TRI 2.0a : Methodological guide and user's documentation. Technical report, LAMSADE, Université Paris-Dauphine, 1999.

[121] W. Yu. ELECTRE TRI: Aspects methodologiques et manuel dútilisation. Technical report, LAMSADE, Université de Paris-Dauphine, 1992.

[122] Y. Chen. Multiple Criteria Decision Analysis: Classification Problems and Solutions. Phd thesis, Systems Design Engineering, University of Waterloo, 2006.

[123] Y. Goletsis, C. Papaloukas, D. Fotiadis, A. Likas, and L. Michalis. Automated ischemic beat classification using genetic algorithms and multicriteria decision analysis. *IEEE Transactions on Biomedical Engineering*, 51(10):1717 – 1725, 2004.

[124] Y. Kodatoff and R.S. Michalski. *Machine Learning: An Artificial Intelligence Approach*, volume 3. Morgan Kaufmann Publishers, 1990.

[125] Y. Kodatoff and R.S. Michalski. *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press, 1990.

Appendix A

Appendix

Tab. A.1: gMCDA performance when applied to the Wisconsin Breast Cancer dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	93.66%	91.6%	92.39%	92.71%	91.28%	91.92%	90.33%	90.8%	91.44%	93.82%	92%	1.09
2	93.66%	91.6%	92.39%	92.71%	91.28%	91.92%	90.33%	90.8%	91.44%	93.82%	92%	1.09
3	93.66%	91.6%	92.39%	92.71%	91.28%	91.92%	90.33%	90.8%	91.44%	93.82%	92%	1.09
4	93.66%	91.6%	92.39%	92.71%	91.28%	91.92%	90.33%	90.8%	91.44%	93.82%	92%	1.09
5	93.66%	91.6%	92.39%	92.71%	91.28%	91.92%	90.33%	90.8%	91.44%	93.82%	92%	1.09
6	94.13%	93.34%	92.07%	91.28%	90.8%	91.76%	91.6%	93.34%	91.76%	93.66%	92.37%	1.08
7	94.13%	93.34%	92.07%	91.28%	90.8%	91.76%	91.6%	93.34%	91.76%	93.66%	92.37%	1.08
8	94.13%	93.34%	92.07%	91.28%	90.8%	91.6%	91.6%	93.34%	91.76%	93.66%	92.36%	1.09
9	94.13%	93.34%	92.07%	91.75%	91.13%	91.92%	91.76%	93.66%	91.92%	93.66%	92.53%	0.99
10	94.13%	93.34%	92.07%	91.28%	90.8%	91.6%	91.6%	93.34%	91.76%	93.66%	92.36%	1.09
11	93.50%	93.34%	91.60%	90.96%	91.13%	90.97%	91.92%	93.82%	91.13%	93.5%	92.19%	1.15
12	93.34%	93.34%	90.96%	90.96%	90.8%	90.81%	91.76%	93.02%	91.13%	93.50%	91.96%	1.12
13	72.11%	77.02%	55.15%	87.01%	87.48%	56.89%	87.96%	78.28%	43.1%	90.49%	73.53%	15.65
14	72.58%	76.54%	56.25%	87.01%	87.63%	57.05%	87.8%	78.28%	42.63%	90.33%	73.61%	15.57
15	72.42%	77.02%	55.15%	87.01%	87.48%	56.58%	87.96%	78.28%	42.63%	90.49%	73.50%	15.77
16	72.26%	76.86%	55.30%	87.01%	88.11%	56.41%	88.11%	78.12%	42.95%	90.64%	73.56%	15.79
17	72.42%	76.70%	55.65%	87.01%	87.63%	57.05%	87.8%	78.44%	42.31%	90.33%	73.53%	15.71
18	72.26%	76.86%	55.78%	87.01%	87.32%	57.21%	87.96%	78.13%	42.79%	90.64%	73.58%	15.6
19	72.42%	77.17%	55.46%	87.01%	87.48%	57.05%	88.11%	78.13%	42.95%	90.64%	73.62%	15.66
20	33.12%	47.22%	31.53%	41.36%	45.25%	29.16%	34.07%	42.79%	32.49%	34.23%	37.12%	6.08
21	32.8%	47.96%	31.53%	41.36%	39.94%	29.16%	34.7%	42.63%	32.33%	34.39%	36.59%	5.49
22	32.64%	47.06%	31.22%	41.36%	39.46%	28.05%	34.7%	42.63%	32.17%	34.55%	36.38%	5.66
23	32.8%	46.75%	31.7%	41.05%	39.77%	28.52%	33.43%	42.31%	32.01%	33.76%	36.21%	5.53
24	32.64%	46.75%	31.85%	40.73%	38.67%	27.41%	34.23%	42.47%	32.49%	34.07%	36.13%	5.55
25	36.13%	36.76%	41.52%	32.65%	29 %	29 %	29.64%	28.52%	34.87%	29.95%	32.80%	4.15
26	31.38%	46.28%	42.47%	28.37%	28.21%	39.93%	35.5%	26.47%	26.94%	28.84%	33.44%	6.79
27	31.38%	46.43%	42.31%	28.52%	28.05%	40.25%	35.34%	26.62%	26.47%	28.68%	33.41%	6.87
28	31.54%	46.59%	42.31%	28.84%	28.37%	39.14%	33.28%	27.1%	26.96%	28.84%	33.3%	6.61
29	31.54%	45.95%	42.47%	28.37%	28.05%	40.89%	35.34%	26.78%	27.1%	28.84%	33.53%	6.79
30	31.38%	46.11%	42%	28.21%	28.21%	38.82%	35.34%	26.62%	27.1%	28.68%	33.25%	6.59
Mean	66.32%	71.65%	63.48%	69.74%	69.12%	66.28%	69.49%	67.68%	57.62%	70.09%		
Std. Dev.	27.5	21.31	24.96	27.5	27.86	26.21	27.4	27.01	28.64	30.05		

Tab. A.2: PROAFTN performance when applied to the Wisconsin Breast Cancer dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	94.00%	91.37%	91.29%	91.08%	92.88%	89.01%	88.20%	88.45%	87.33%	92.16%	90.58%	2.22
2	90.91%	88.05%	87.34%	89.37%	92.31%	89.44%	89.65%	87.18%	90.02%	88.87%	89.31%	1.58
3	93.23%	90.04%	87.97%	92.79%	93.65%	88.15%	90.89%	90.55%	89.25%	94.23%	91.07%	2.29
4	87.23%	90.93%	88.80%	90.89%	92.31%	87.72%	92.34%	83.19%	93.09%	92.16%	89.87%	3.11
5	94.78%	92.92%	95.44%	93.74%	94.42%	93.75%	92.13%	91.18%	90.21%	94.02%	93.26%	1.65
6	94.20%	92.48%	89.42%	90.89%	93.85%	90.30%	91.72%	92.23%	90.02%	91.55%	91.66%	1.58
7	88.97%	90.04%	81.12%	92.60%	91.54%	83.62%	90.06%	90.97%	87.14%	82.89%	87.90%	4.02
8	93.23%	91.15%	89.00%	92.41%	94.62%	87.72%	92.75%	89.50%	92.90%	91.75%	91.50%	2.16
9	95.55%	90.71%	92.12%	92.22%	94.81%	90.09%	93.37%	93.91%	92.32%	95.05%	93.01%	1.84
10	92.07%	90.04%	87.34%	90.70%	92.69%	88.58%	89.03%	90.34%	85.22%	91.55%	89.76%	2.28
11	93.42%	92.48%	90.04%	93.93%	92.50%	87.07%	86.75%	91.39%	94.43%	94.23%	91.62%	2.82
12	93.04%	89.16%	90.25%	88.80%	93.46%	89.01%	89.23%	91.81%	89.64%	92.16%	90.66%	1.79
13	89.17%	93.58%	91.91%	92.22%	93.46%	85.99%	91.72%	88.03%	92.32%	92.37%	91.08%	2.50
14	95.74%	91.15%	92.12%	88.61%	92.12%	84.27%	87.37%	90.34%	90.02%	90.10%	90.18%	3.07
15	93.23%	89.38%	87.14%	90.13%	91.73%	87.93%	91.51%	90.13%	91.17%	90.10%	90.25%	1.81
16	93.42%	91.37%	91.08%	91.65%	94.04%	88.15%	92.34%	90.76%	91.55%	93.81%	91.82%	1.74
17	95.55%	92.26%	90.66%	92.41%	94.04%	89.66%	91.93%	91.60%	91.75%	91.55%	92.14%	1.65
18	92.46%	87.39%	87.34%	91.46%	92.88%	88.58%	89.23%	87.82%	88.48%	92.37%	89.80%	2.24
19	93.23%	91.37%	90.04%	93.36%	92.50%	88.79%	91.51%	89.29%	90.40%	91.55%	91.20%	1.57
20	92.84%	86.95%	88.80%	89.18%	91.92%	82.76%	89.86%	89.71%	85.41%	90.52%	88.79%	3.03
21	93.42%	91.15%	92.74%	93.74%	93.08%	90.52%	90.06%	89.71%	90.79%	93.61%	91.88%	1.58
22	94.20%	93.36%	89.21%	93.93%	93.65%	88.58%	91.51%	88.24%	91.55%	92.37%	91.66%	2.27
23	93.42%	88.94%	90.04%	93.74%	93.46%	82.97%	90.48%	90.13%	93.67%	90.31%	90.72%	3.28
24	94.58%	91.15%	92.32%	90.32%	93.46%	83.84%	88.82%	91.60%	93.47%	92.58%	91.21%	3.09
25	94.39%	89.38%	86.93%	86.72%	92.50%	87.28%	83.44%	87.82%	86.18%	86.80%	88.14%	3.19
26	93.04%	94.69%	95.64%	93.55%	94.04%	94.40%	95.45%	91.39%	93.09%	94.43%	93.97%	1.26
27	93.04%	92.48%	89.63%	92.03%	92.50%	90.09%	92.75%	92.86%	93.47%	91.75%	92.06%	1.26
28	93.42%	92.70%	92.74%	93.74%	94.62%	92.67%	90.89%	90.13%	93.28%	94.43%	92.86%	1.42
29	93.62%	92.26%	87.55%	91.27%	93.08%	88.79%	87.78%	90.34%	89.25%	91.96%	90.59%	2.18
30	94.58%	91.81%	92.74%	93.17%	92.88%	91.38%	91.72%	86.13%	87.33%	92.58%	91.43%	2.65
Mean	93.13%	91.03%	89.96%	91.69%	93.17%	88.37%	90.48%	89.89%	90.49%	91.79%		
Std. Dev.	1.90	1.83	2.86	1.86	0.89	2.89	2.34	2.14	2.58	2.45		

Tab. A.3: gMCDA performance when applied to the Iris dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	66.66%	62%	56%	60.66%	70.66%	66.66%	60.66%	68%	60%	73.33%	64.46%	5.15
2	66.66%	62%	56%	60.66%	70.66%	66.66%	60.66%	68%	60%	73.33%	64.46%	5.15
3	66.66%	62%	56%	60.66%	70.66%	66.66%	60.66%	68%	60%	73.33%	64.46%	5.15
4	66.66%	62%	56%	60.66%	70.66%	66.66%	60.66%	68%	60%	73.33%	64.46%	5.15
5	66.66%	62%	56%	60.66%	70.66%	66.66%	60.66%	68%	60%	73.33%	64.46%	5.15
6	74%	73.33%	68%	75.33%	74.66%	83.33%	75.33%	82.66%	68.66%	80%	75.53%	4.93
7	74%	73.33%	68%	75.33%	74.66%	83.33%	75.33%	82.66%	68.66%	80%	75.53%	4.93
8	74%	73.33%	68%	75.33%	74.66%	83.33%	75.33%	82.66%	68.66%	80%	75.53%	4.93
9	74%	73.33%	68%	75.33%	74.66%	83.33%	75.33%	82.66%	68.66%	80%	75.53%	4.93
10	74%	73.33%	68%	75.33%	74.66%	83.33%	75.33%	82.66%	68.66%	80%	75.53%	4.93
11	72.66%	72.66%	68%	74.66%	74.66%	82%	75.33%	82%	68.66%	80%	75.06%	4.72
12	72%	72.66%	68%	74.66%	73.33%	82%	75.33%	82%	68.66%	80%	74.86%	4.79
13	85.33%	68.66%	72.66%	75.22%	83.33%	88%	79.33%	83.33%	66%	85.33%	78.73%	7.26
14	85.33%	68.66%	72.66%	75.22%	83.33%	88%	79.33%	83.33%	66%	85.33%	78.73%	7.26
15	85.33%	68.66%	72.66%	75.22%	83.33%	88%	79.33%	83.33%	66%	85.33%	78.73%	7.26
16	85.33%	68.66%	72.66%	75.22%	83.33%	88%	79.33%	83.33%	66%	85.33%	78.73%	7.26
17	85.33%	68.66%	72.66%	75.22%	83.33%	88%	79.33%	83.33%	66%	85.33%	78.73%	7.26
18	85.33%	68.66%	72.66%	75.22%	83.33%	88%	79.33%	83.33%	66%	85.33%	78.73%	7.26
19	85.33%	68.66%	72.66%	75.22%	83.33%	88%	79.33%	83.33%	66%	85.33%	78.73%	7.26
20	82%	70%	78%	80.66%	80%	87.33%	84.66%	84%	74.66%	88%	80.93%	5.32
21	82%	69.33%	77.33%	78.66%	80%	88%	84%	82.66%	74%	88%	80.40%	5.58
22	82%	70%	78%	78.66%	80%	86.66%	84.66%	83.33%	72.66%	87.33%	80.33%	5.41
23	82%	70%	78%	80%	80%	87.33%	84.66%	84%	74.66%	88%	80.87%	5.33
24	82%	70%	78%	80%	80%	87.33%	84.66%	84%	75.33%	88%	80.93%	5.25
25	78.66%	67.33%	72.66%	72%	75.22%	76.66%	76%	74.66%	68.66%	81.33%	74.33%	4.07
26	78.66%	63.33%	66%	70.66%	74.66%	81.33%	73.33%	76%	65.33%	86.66%	73.6%	7.09
27	79.33%	62.66%	67.33%	70%	74.66%	81.33%	72%	75.33%	65.33%	86.66%	73.46%	7.14
28	78.66%	63.33%	66%	70.66%	74.66%	81.33%	73.33%	76%	65.33%	86.66%	73.6%	7.09
29	78%	62.66%	66%	70.66%	74.66%	81.33%	73.33%	76%	66%	86.66%	73.53%	7.07
30	77.33%	64%	66%	70.66%	74.66%	80.66%	72.66%	75.33%	65.33%	86.66%	73.33%	6.82
Mean	77.53%	67.84%	68.6%	72.6%	76.93%	81.64%	74.97%	79.06%	67%	82.6%		
Std. Dev.	6.54	4.16	6.92	6.09	4.53	7.45	7.51	5.88	4.37	5.09		

Tab. A.4: PROAFTN performance when applied to the Iris dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	83.82%	82.22%	75.36%	79.56%	81.16%	76.81%	80.29%	77.54%	67.65%	82.61%	78.70%	4.73
2	91.91%	83.70%	87.68%	79.56%	78.26%	92.75%	94.16%	78.26%	75.74%	89.86%	85.19%	6.91
3	91.18%	65.93%	69.57%	75.91%	74.64%	87.68%	80.29%	76.81%	86.76%	88.41%	79.72%	8.58
4	80.88%	57.78%	59.42%	75.18%	77.54%	86.23%	84.67%	78.99%	65.44%	87.68%	75.38%	10.89
5	90.44%	79.26%	86.96%	78.10%	87.68%	90.58%	91.24%	89.13%	80.88%	89.13%	86.34%	4.99
6	97.06%	86.67%	86.23%	90.51%	92.75%	95.65%	91.24%	92.03%	75.74%	90.58%	89.85%	6.00
7	86.76%	76.30%	77.54%	83.94%	73.19%	89.86%	75.91%	77.54%	80.15%	83.33%	80.45%	5.34
8	74.26%	71.85%	74.64%	78.10%	72.46%	80.43%	83.21%	75.36%	64.71%	74.64%	74.97%	5.05
9	84.56%	62.22%	65.22%	81.02%	73.91%	76.81%	81.02%	84.78%	68.38%	86.96%	76.49%	8.74
10	86.76%	80.74%	81.16%	89.05%	92.75%	87.68%	83.94%	91.30%	74.26%	85.51%	85.32%	5.53
11	89.71%	83.70%	80.43%	87.59%	84.78%	86.96%	80.29%	86.23%	80.15%	90.58%	85.04%	3.85
12	91.18%	86.67%	81.88%	91.24%	80.43%	82.61%	94.89%	94.93%	87.50%	87.68%	87.90%	5.18
13	84.56%	63.70%	69.57%	73.72%	76.09%	84.78%	80.29%	77.54%	71.32%	92.03%	77.36%	8.36
14	80.15%	80.00%	81.16%	77.37%	89.86%	87.68%	81.75%	89.13%	89.71%	92.03%	84.88%	5.28
15	89.71%	84.44%	78.26%	86.86%	71.01%	77.54%	91.24%	90.58%	75.00%	82.61%	82.73%	7.05
16	89.71%	71.85%	72.46%	78.83%	91.30%	86.96%	90.51%	79.71%	75.74%	90.58%	82.77%	7.88
17	92.65%	82.22%	82.61%	93.43%	89.13%	92.03%	89.78%	89.13%	82.35%	90.58%	88.39%	4.38
18	85.29%	73.33%	78.26%	77.37%	81.16%	86.23%	90.51%	94.20%	80.88%	89.13%	83.64%	6.55
19	83.09%	78.52%	68.84%	81.02%	81.16%	97.10%	91.24%	86.23%	83.82%	86.96%	83.80%	7.56
20	81.62%	80.00%	74.64%	81.75%	73.19%	86.96%	86.13%	78.26%	77.94%	84.78%	80.53%	4.65
21	80.88%	78.52%	65.22%	84.67%	76.09%	86.23%	87.59%	91.30%	72.79%	85.51%	80.88%	7.87
22	85.29%	77.04%	74.64%	78.83%	83.33%	86.96%	86.86%	80.43%	78.68%	95.65%	82.77%	6.18
23	80.88%	76.30%	90.58%	82.48%	92.03%	84.78%	89.05%	91.30%	78.68%	83.33%	84.94%	5.56
24	91.91%	94.81%	70.29%	85.40%	90.58%	84.78%	94.16%	90.58%	80.15%	92.03%	87.47%	7.59
25	80.15%	74.81%	79.71%	81.02%	94.20%	84.78%	81.75%	84.06%	67.65%	86.96%	81.51%	7.07
26	85.29%	79.26%	74.64%	82.48%	84.78%	85.51%	95.62%	92.75%	88.24%	89.13%	85.77%	6.16
27	79.41%	69.63%	59.42%	78.83%	83.33%	82.61%	89.05%	79.71%	83.82%	79.71%	78.55%	8.34
28	92.65%	80.00%	82.61%	92.70%	87.68%	84.78%	82.48%	81.88%	72.79%	86.23%	84.38%	5.95
29	88.24%	78.52%	78.26%	83.21%	81.16%	91.30%	96.35%	84.78%	73.53%	81.16%	83.65%	6.77
30	78.68%	85.19%	71.74%	73.72%	77.54%	78.99%	89.05%	83.33%	70.59%	85.51%	79.43%	6.25
Mean	85.96%	77.51%	75.97%	82.12%	82.44%	86.14%	87.15%	84.93%	77.03%	87.03%		
Std. Dev.	5.32	7.96	7.85	5.48	7.06	5.00	5.53	6.08	6.83	4.33		

Tab. A.5: gMCDA performance when applied to the Zoo dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	93.07%	90.1%	91.09%	93.07%	96.03%	93.07%	96.04%	90.1%	94.06%	98.02%	93.47%	2.51
2	93.07%	90.1%	91.09%	93.07%	96.03%	93.07%	96.04%	90.1%	94.06%	98.02%	93.47%	2.51
3	93.07%	90.1%	91.09%	93.07%	96.03%	93.07%	96.04%	90.1%	94.06%	98.02%	93.47%	2.51
4	93.07%	90.1%	91.09%	93.07%	96.03%	93.07%	96.04%	90.1%	94.06%	98.02%	93.47%	2.51
5	93.07%	90.1%	91.09%	93.07%	96.03%	93.07%	96.04%	90.1%	94.06%	98.02%	93.47%	2.51
6	0%	0%	0%	0%	0%	0%	1%	0%	0%	0%	0.1%	0.3
7	0%	0%	0%	0%	0%	0%	1%	0%	0%	0%	0.1%	0.3
8	0%	0%	1%	0%	0%	0%	1%	0%	0%	0%	0.2%	0.4
9	0%	1%	0%	0%	0%	0%	1%	0%	0%	0%	0.2%	0.4
10	0%	0%	1%	0%	0%	0%	1%	0%	0%	0%	0.2%	0.4
11	0%	0%	1%	0%	0%	0%	1%	0%	0%	0%	0.2%	0.4
12	0%	0%	1%	0%	0%	0%	1%	0%	0%	0%	0.2%	0.4
13	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
14	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
15	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
16	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
17	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
18	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
19	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
20	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
21	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
22	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
23	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
24	40.6%	40.6%	6.93%	40.59%	40.59%	40.59%	40.59%	40.59%	6.93%	40.59%	33.86%	13.46
25	40.6%	40.6%	37.62%	40.59%	40.59%	40.59%	40.59%	40.59%	37.62%	40.59%	40%	1.18
26	4.95%	4.95%	4.95%	33.66%	4.95%	2.97%	4.95%	33.66%	2.97%	4.95%	10.3%	11.7
27	4.95%	4.95%	4.95%	33.66%	4.95%	2.97%	4.95%	33.66%	2.97%	4.95%	10.3%	11.7
28	4.95%	4.95%	4.95%	33.66%	4.95%	2.97%	4.95%	33.66%	2.97%	4.95%	10.3%	11.7
29	4.95%	4.95%	4.95%	33.66%	4.95%	2.97%	4.95%	33.66%	2.97%	4.95%	10.3%	11.7
30	4.95%	4.95%	4.95%	33.66%	4.95%	2.97%	4.95%	33.66%	2.97%	4.95%	10.3%	11.7
Mean	23.33%	31.32%	22.93%	18.18%	23.21%	36.7%	17.39%	37.02%	31.45%	35.9%		
Std. Dev.	26.9	24.84	26.78	25.4	27.21	23.56	24.99	23.04	25.65	22.74		

Tab. A.6: PROAFTN performance when applied to the Zoo dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	72.50%	64.63%	72.73%	74.07%	62.35%	72.00%	62.67%	73.61%	65.22%	61.04%	68.08%	5.32
2	66.25%	73.17%	67.53%	64.20%	43.53%	78.67%	54.67%	66.67%	68.12%	62.34%	64.51%	9.70
3	62.50%	51.22%	48.05%	49.38%	67.06%	69.33%	50.67%	56.94%	71.01%	58.44%	58.46%	8.64
4	70.00%	54.88%	64.94%	71.60%	64.71%	72.00%	73.33%	58.33%	69.57%	63.64%	66.30%	6.15
5	68.75%	68.29%	63.64%	62.96%	74.12%	65.33%	68.00%	48.61%	59.42%	64.94%	64.41%	6.83
6	66.25%	67.07%	46.75%	62.96%	40.00%	53.33%	46.67%	45.83%	60.87%	46.75%	53.65%	9.83
7	72.50%	69.51%	51.95%	51.85%	42.35%	53.33%	56.00%	47.22%	66.67%	23.38%	53.48%	14.43
8	65.00%	75.61%	59.74%	67.90%	63.53%	52.00%	56.00%	63.89%	49.28%	57.14%	61.01%	7.83
9	71.25%	74.39%	50.65%	62.96%	29.41%	60.00%	50.67%	48.61%	73.91%	62.34%	58.42%	13.97
10	63.75%	64.63%	71.43%	79.01%	55.29%	61.33%	54.67%	63.89%	59.42%	50.65%	62.41%	8.33
11	56.25%	74.39%	72.73%	76.54%	75.29%	64.00%	64.00%	70.83%	63.77%	74.03%	69.18%	6.73
12	81.25%	71.95%	68.83%	76.54%	62.35%	74.67%	72.00%	56.94%	68.12%	74.03%	70.67%	7.03
13	58.75%	58.54%	57.14%	56.79%	57.65%	60.00%	41.33%	50.00%	52.17%	46.75%	53.91%	6.17
14	65.00%	52.44%	54.55%	48.15%	41.18%	62.67%	56.00%	68.06%	68.12%	58.44%	57.46%	8.81
15	75.00%	64.63%	80.52%	77.78%	62.35%	80.00%	76.00%	62.50%	57.97%	74.03%	71.08%	8.33
16	72.50%	69.51%	59.74%	65.43%	48.24%	70.67%	45.33%	63.89%	59.42%	70.13%	62.49%	9.43
17	55.00%	62.20%	62.34%	60.49%	78.82%	44.00%	42.67%	37.50%	53.62%	38.96%	53.56%	13.00
18	68.75%	60.98%	70.13%	67.90%	77.65%	56.00%	69.33%	68.06%	65.22%	58.44%	66.25%	6.34
19	73.75%	75.61%	62.34%	56.79%	61.18%	61.33%	42.67%	79.17%	69.57%	46.75%	62.91%	12.03
20	77.50%	67.07%	68.83%	81.48%	67.06%	64.00%	58.67%	66.67%	75.36%	74.03%	70.07%	6.89
21	51.25%	65.85%	62.34%	62.96%	62.35%	54.67%	66.67%	59.72%	53.62%	48.05%	58.75%	6.42
22	75.00%	81.71%	75.32%	70.37%	75.29%	76.00%	48.00%	72.22%	68.12%	59.74%	70.18%	9.73
23	68.75%	71.95%	77.92%	66.67%	76.47%	76.00%	66.67%	62.50%	50.72%	64.94%	68.26%	8.11
24	65.00%	59.76%	62.34%	61.73%	61.18%	60.00%	53.33%	52.78%	60.87%	49.35%	58.63%	5.02
25	70.00%	62.20%	54.55%	56.79%	62.35%	49.33%	49.33%	70.83%	71.01%	55.84%	60.22%	8.39
26	63.75%	70.73%	71.43%	81.48%	58.82%	69.33%	73.33%	72.22%	63.77%	67.53%	69.24%	6.27
27	76.25%	73.17%	59.74%	53.09%	47.06%	76.00%	73.33%	63.89%	75.36%	80.52%	67.84%	11.27
28	63.75%	64.63%	62.34%	51.85%	70.59%	69.33%	60.00%	66.67%	50.72%	50.65%	61.05%	7.55
29	53.75%	59.76%	50.65%	70.37%	50.59%	70.67%	68.00%	69.44%	78.26%	76.62%	64.81%	10.37
30	70.00%	71.95%	50.65%	56.79%	55.29%	52.00%	44.00%	62.50%	66.67%	49.35%	57.92%	9.45
Mean	67.33%	66.75%	62.73%	64.90%	59.80%	64.27%	58.13%	61.67%	63.86%	58.96%		
Std. Dev.	7.26	7.29	9.16	9.66	12.59	9.59	10.73	9.72	7.98	12.56		

Tab. A.7: gMCDA performance when applied to the Balance dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	54.88%	61.44%	60.32%	56.48%	59.52%	58.87%	62.56%	60.96%	60.16%	56.16%	59.14%	2.52
2	54.88%	61.44%	60.32%	56.48%	59.52%	58.87%	62.56%	60.96%	60.16%	56.16%	59.14%	2.52
3	54.88%	61.44%	60.32%	56.48%	59.52%	58.87%	62.56%	60.96%	60.16%	56.16%	59.14%	2.52
4	54.88%	61.44%	60.32%	56.48%	59.52%	58.87%	62.56%	60.96%	60.16%	56.16%	59.14%	2.52
5	54.88%	61.44%	60.32%	56.48%	59.52%	58.87%	62.56%	60.96%	60.16%	56.16%	59.14%	2.52
6	60.00%	67.04%	65.28%	59.84%	66.72%	63.20%	66.88%	63.36%	60.48%	59.52%	63.23%	3.12
7	60.00%	67.04%	65.28%	59.84%	66.72%	63.20%	66.88%	63.36%	60.48%	59.52%	63.23%	3.12
8	60.00%	67.04%	65.28%	59.84%	66.72%	63.20%	66.88%	63.36%	60.48%	59.52%	63.23%	3.12
9	61.44%	67.36%	65.92%	60.16%	67.68%	64.32%	67.36%	64.00%	61.44%	61.12%	64.08%	2.91
10	59.84%	67.04%	65.28%	59.84%	66.88%	63.20%	66.88%	63.36%	60.48%	59.52%	63.23%	3.16
11	61.28%	68.48%	66.88%	60.32%	67.04%	63.36%	66.88%	63.52%	60.64%	61.76%	64.02%	3.05
12	60.00%	67.04%	65.28%	59.84%	66.72%	63.20%	66.88%	63.36%	44.83%	59.52%	61.67%	6.62
13	43.68%	42.24%	40.96%	32.64%	40.96%	30.08%	37.60%	45.44%	60.48%	36.16%	41.02%	8.38
14	44.80%	41.92%	42.88%	34.08%	42.24%	32.16%	37.44%	45.44%	46.72%	37.28%	40.50%	4.98
15	42.88%	41.44%	40.96%	32.16%	41.44%	28.48%	36.16%	44.64%	46.24%	35.52%	38.99%	5.70
16	42.56%	43.33%	42.40%	35.04%	43.20%	32.64%	37.12%	46.24%	46.56%	37.60%	40.67%	4.76
17	44.16%	41.60%	42.08%	33.12%	42.08%	31.04%	36.80%	44.96%	48.16%	36.64%	40.06%	5.47
18	42.88%	41.44%	40.96%	32.16%	41.44%	28.48%	36.16%	44.64%	46.24%	35.52%	38.99%	5.70
19	42.56%	43.36%	42.40%	35.04%	43.20%	32.64%	37.12%	46.24%	46.56%	37.60%	40.67%	4.76
20	26.56%	27.20%	27.52%	27.84%	26.40%	24.80%	25.76%	29.92%	34.88%	24.64%	27.55%	3.00
21	26.56%	27.20%	27.52%	27.84%	26.40%	24.80%	25.76%	29.92%	34.88%	24.64%	27.55%	3.00
22	26.88%	27.68%	28.48%	27.52%	26.40%	24.96%	25.60%	29.44%	35.20%	24.64%	27.68%	3.05
23	26.40%	27.36%	27.52%	27.84%	26.40%	24.48%	25.60%	30.08%	34.24%	24.48%	27.44%	2.91
24	26.40%	27.52%	28.32%	27.84%	26.88%	24.96%	26.08%	29.76%	35.68%	24.48%	27.79%	3.18
25	24.64%	28.48%	26.72%	27.68%	26.40%	22.08%	24.00%	28.80%	28.00%	25.12%	26.19%	2.19
26	11.04%	17.92%	14.24%	16.32%	13.12%	17.28%	19.68%	15.20%	14.24%	11.84%	15.09%	2.74
27	10.88%	18.40%	13.76%	16.16%	13.28%	17.44%	20.16%	15.36%	14.39%	11.52%	15.14%	2.97
28	10.88%	18.40%	14.24%	15.84%	13.76%	17.44%	20.00%	15.36%	14.24%	11.68%	15.18%	2.86
29	10.72%	17.92%	14.40%	16.48%	13.44%	18.40%	19.68%	15.36%	14.88%	11.84%	15.31%	2.88
30	10.72%	18.24%	14.24%	16.32%	13.12%	17.76%	19.68%	15.52%	13.92%	11.84%	15.14%	2.89
Mean	40.41%	44.33%	43.01%	39.47%	42.87%	39.60%	43.06%	44.05%	44.17%	38.81%		
Std. Dev.	18.08	18.84	19.17	16.79	19.84	18.74	19.22	18.05	16.98	18.17		

Tab. A.8: PROAFTN performance when applied to the Balance dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	65.84%	60.88%	69.20%	60.71%	63.89%	63.54%	66.55%	74.34%	68.67%	69.38%	66.30%	4.25
2	55.04%	58.41%	50.27%	51.86%	57.70%	59.12%	61.77%	64.42%	67.61%	55.93%	58.21%	5.37
3	68.67%	67.79%	68.50%	60.00%	64.07%	70.62%	67.26%	66.55%	67.96%	69.38%	67.08%	3.04
4	66.37%	70.80%	64.07%	63.36%	63.72%	66.02%	69.56%	69.38%	65.49%	71.68%	67.04%	3.07
5	66.37%	68.67%	62.48%	70.09%	63.54%	64.25%	68.50%	73.10%	68.50%	64.60%	67.01%	3.34
6	48.32%	61.06%	46.73%	46.02%	56.11%	52.39%	50.09%	54.16%	57.52%	48.85%	52.12%	5.00
7	60.71%	61.95%	59.82%	62.48%	58.94%	64.25%	62.30%	63.19%	66.90%	57.88%	61.84%	2.65
8	64.25%	65.66%	64.96%	62.30%	67.61%	66.02%	63.72%	77.35%	71.15%	65.84%	66.88%	4.39
9	65.49%	64.42%	56.99%	63.36%	65.13%	59.12%	64.60%	71.15%	68.85%	64.42%	64.35%	4.10
10	65.84%	62.83%	63.19%	64.60%	63.54%	62.83%	67.43%	72.57%	68.14%	65.31%	65.63%	3.07
11	63.54%	60.00%	58.58%	64.78%	58.76%	63.72%	60.53%	66.73%	69.03%	61.42%	62.71%	3.47
12	71.33%	63.19%	54.34%	65.31%	63.01%	63.01%	66.73%	70.80%	67.79%	62.65%	64.81%	4.88
13	56.81%	59.82%	48.50%	62.65%	57.52%	65.66%	66.55%	66.37%	67.61%	67.43%	61.89%	6.22
14	65.13%	58.76%	60.53%	55.93%	64.25%	61.42%	60.35%	70.27%	71.86%	63.01%	63.15%	4.95
15	69.56%	68.85%	65.31%	68.67%	68.32%	76.64%	71.15%	65.31%	68.85%	68.85%	69.15%	3.18
16	74.69%	67.96%	72.57%	67.79%	73.81%	74.69%	70.97%	73.10%	77.52%	72.74%	72.58%	3.02
17	69.73%	62.48%	64.78%	60.00%	60.71%	62.12%	63.72%	69.73%	68.85%	70.62%	65.27%	4.09
18	42.30%	55.75%	48.85%	51.33%	54.51%	46.19%	49.91%	58.58%	51.68%	39.29%	49.84%	5.97
19	55.22%	63.89%	55.04%	59.12%	57.70%	55.58%	63.19%	63.36%	58.23%	57.17%	58.85%	3.46
20	77.88%	75.75%	77.70%	66.55%	73.10%	77.17%	71.86%	72.39%	70.27%	72.39%	73.50%	3.64
21	67.96%	66.37%	64.96%	67.08%	67.43%	65.66%	65.84%	72.74%	64.60%	65.66%	66.83%	2.33
22	50.09%	55.04%	52.74%	51.68%	54.16%	54.69%	54.87%	71.50%	64.25%	58.23%	56.73%	6.49
23	63.36%	60.88%	65.49%	60.00%	67.08%	64.25%	67.96%	75.04%	66.55%	70.09%	66.07%	4.43
24	69.91%	69.38%	62.65%	67.96%	62.83%	64.42%	71.33%	72.04%	70.27%	67.61%	67.84%	3.43
25	65.31%	67.08%	67.43%	62.12%	62.65%	63.19%	71.86%	62.30%	67.61%	62.65%	65.22%	3.22
26	69.38%	67.79%	67.96%	70.44%	67.79%	63.72%	68.14%	72.04%	70.62%	70.09%	68.80%	2.30
27	65.66%	62.65%	66.55%	60.71%	59.29%	57.70%	64.78%	72.74%	69.56%	70.09%	64.97%	4.94
28	66.19%	67.43%	64.42%	64.60%	68.14%	68.50%	68.85%	70.44%	70.27%	64.78%	67.36%	2.27
29	59.82%	61.24%	61.42%	65.84%	62.65%	62.48%	60.53%	66.19%	67.96%	64.96%	63.31%	2.75
30	59.12%	65.13%	57.88%	65.66%	62.48%	60.35%	59.47%	69.91%	67.43%	64.60%	63.20%	3.98
Mean	63.66%	64.06%	61.46%	62.10%	63.01%	63.31%	64.68%	68.93%	67.39%	64.25%		
Std. Dev.	7.71	4.63	7.36	5.87	4.92	6.53	5.75	5.11	4.70	7.21		

Tab. A.9: gMCDA performance when applied to the Lung Cancer dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	55.17%	51.72%	55.17%	62.07%	51.72%	48.28%	51.72%	48.28%	37.93%	55.17%	51.72%	6.30
2	31.03%	48.28%	58.62%	41.38%	55.17%	48.28%	51.72%	37.93%	48.28%	44.83%	46.55%	8.17
3	31.03%	41.38%	44.83%	44.83%	37.93%	37.93%	51.72%	44.83%	31.03%	37.93%	40.34%	6.51
4	37.93%	41.38%	58.62%	55.17%	51.72%	48.28%	55.17%	37.93%	44.83%	48.28%	47.93%	7.35
5	51.72%	58.62%	51.72%	41.38%	51.72%	51.72%	51.72%	55.17%	34.48%	58.62%	50.69%	7.46
6	48.28%	41.38%	41.38%	48.28%	51.72%	48.28%	48.28%	41.38%	41.38%	44.83%	45.52%	3.91
7	37.93%	41.38%	41.38%	48.28%	55.17%	44.83%	55.17%	51.72%	55.17%	51.72%	48.28%	6.50
8	41.38%	48.28%	48.28%	51.72%	44.83%	55.17%	48.28%	44.83%	41.38%	55.17%	47.93%	5.00
9	41.38%	44.83%	41.38%	41.38%	41.38%	41.38%	48.28%	31.03%	37.93%	44.83%	41.38%	4.60
10	27.59%	44.83%	51.72%	55.17%	58.62%	51.72%	55.17%	44.83%	51.72%	37.93%	47.93%	9.40
11	20.69%	37.93%	37.93%	41.38%	48.28%	37.93%	34.48%	37.93%	37.93%	37.93%	37.24%	6.86
12	51.72%	41.38%	51.72%	48.28%	48.28%	48.28%	51.72%	37.93%	44.83%	44.83%	46.90%	4.65
13	31.03%	41.38%	55.17%	41.38%	48.28%	34.48%	44.83%	24.14%	41.38%	31.03%	39.31%	9.22
14	34.48%	44.83%	48.28%	37.93%	44.83%	44.83%	44.83%	37.93%	48.28%	48.28%	43.45%	4.93
15	44.83%	44.83%	48.28%	44.83%	55.17%	44.83%	41.38%	44.83%	44.83%	51.72%	46.55%	4.06
16	51.72%	34.48%	44.83%	51.72%	51.72%	58.62%	48.28%	37.93%	41.38%	58.62%	47.93%	8.20
17	37.93%	51.72%	48.28%	48.28%	55.17%	37.93%	44.83%	44.83%	55.17%	48.28%	47.24%	6.09
18	44.83%	34.48%	48.28%	48.28%	37.93%	51.72%	65.52%	55.17%	41.38%	41.38%	46.90%	9.08
19	34.48%	27.59%	34.48%	24.14%	37.93%	44.83%	34.48%	31.03%	37.93%	41.38%	34.83%	6.18
20	27.59%	44.83%	44.83%	44.83%	48.28%	44.83%	44.83%	41.38%	48.28%	51.72%	44.14%	6.46
21	44.83%	34.48%	48.28%	55.17%	51.72%	51.72%	55.17%	44.83%	62.07%	62.07%	51.03%	8.42
22	48.28%	44.83%	44.83%	48.28%	48.28%	41.38%	51.72%	51.72%	48.28%	48.28%	47.59%	3.17
23	44.83%	34.48%	44.83%	51.72%	41.38%	44.83%	55.17%	48.28%	41.38%	48.28%	45.52%	5.82
24	44.83%	37.93%	51.72%	37.93%	41.38%	34.48%	48.28%	44.83%	37.93%	41.38%	42.07%	5.34
25	34.48%	31.03%	27.59%	27.59%	27.59%	41.38%	34.48%	24.14%	31.03%	34.48%	31.38%	5.00
26	44.83%	48.28%	48.28%	44.83%	48.28%	51.72%	62.07%	37.93%	44.83%	44.83%	47.59%	6.25
27	37.93%	48.28%	48.28%	55.17%	58.62%	48.28%	58.62%	44.83%	48.28%	58.62%	50.69%	6.91
28	41.38%	44.83%	51.72%	51.72%	37.93%	48.28%	51.72%	51.72%	48.28%	48.28%	47.59%	4.82
29	37.93%	41.38%	44.83%	48.28%	41.38%	48.28%	44.83%	34.48%	37.93%	44.83%	42.41%	4.61
30	37.93%	44.83%	55.17%	41.38%	44.83%	41.38%	55.17%	55.17%	44.83%	44.83%	46.55%	6.35
Mean	40.00%	42.53%	47.36%	46.09%	47.24%	45.86%	49.66%	42.30%	43.68%	47.01%		
Std. Dev.	8.23	6.61	6.77	7.97	7.25	5.88	7.39	8.30	7.03	7.49		

Tab. A.10: PROAFTN performance when applied to the Lung Cancer dataset with random prototypes.

Param	1	2	3	4	5	6	7	8	9	10	Mean	Std. Dev.
1	31.03%	34.48%	48.28%	51.72%	37.93%	37.93%	41.38%	48.28%	34.48%	48.28%	41.38%	7.27
2	44.83%	44.83%	58.62%	37.93%	44.83%	37.93%	51.72%	41.38%	48.28%	37.93%	44.83%	6.70
3	44.83%	31.03%	41.38%	34.48%	34.48%	44.83%	44.83%	37.93%	34.48%	31.03%	37.93%	5.63
4	44.83%	44.83%	41.38%	44.83%	51.72%	48.28%	37.93%	44.83%	27.59%	51.72%	43.79%	7.09
5	41.38%	34.48%	34.48%	31.03%	24.14%	37.93%	37.93%	34.48%	24.14%	27.59%	32.76%	5.92
6	48.28%	41.38%	44.83%	44.83%	48.28%	37.93%	55.17%	48.28%	48.28%	48.28%	46.55%	4.67
7	34.48%	31.03%	44.83%	37.93%	37.93%	44.83%	41.38%	41.38%	37.93%	34.48%	38.62%	4.54
8	41.38%	48.28%	48.28%	48.28%	62.07%	55.17%	58.62%	62.07%	44.83%	55.17%	52.41%	7.23
9	37.93%	34.48%	44.83%	37.93%	41.38%	44.83%	44.83%	41.38%	44.83%	37.93%	41.03%	3.79
10	31.03%	37.93%	34.48%	24.14%	37.93%	44.83%	31.03%	31.03%	37.93%	24.14%	33.45%	6.51
11	37.93%	20.69%	31.03%	34.48%	37.93%	41.38%	24.14%	27.59%	34.48%	34.48%	32.41%	6.54
12	20.69%	37.93%	44.83%	41.38%	34.48%	24.14%	37.93%	27.59%	27.59%	24.14%	32.07%	8.30
13	27.59%	24.14%	37.93%	37.93%	37.93%	31.03%	44.83%	27.59%	37.93%	37.93%	34.48%	6.50
14	24.14%	44.83%	37.93%	20.69%	37.93%	37.93%	44.83%	44.83%	41.38%	20.69%	35.52%	9.89
15	24.14%	41.38%	41.38%	37.93%	41.38%	37.93%	48.28%	41.38%	34.48%	34.48%	38.28%	6.39
16	27.59%	34.48%	27.59%	34.48%	31.03%	44.83%	31.03%	31.03%	37.93%	51.72%	35.17%	7.76
17	31.03%	34.48%	41.38%	31.03%	34.48%	31.03%	41.38%	34.48%	20.69%	27.59%	32.76%	6.14
18	24.14%	44.83%	24.14%	31.03%	41.38%	41.38%	41.38%	34.48%	37.93%	37.93%	35.86%	7.31
19	34.48%	41.38%	34.48%	41.38%	31.03%	37.93%	24.14%	27.59%	24.14%	44.83%	34.14%	7.35
20	51.72%	44.83%	31.03%	44.83%	37.93%	37.93%	51.72%	55.17%	31.03%	44.83%	43.10%	8.49
21	37.93%	51.72%	37.93%	44.83%	44.83%	37.93%	37.93%	34.48%	34.48%	37.93%	40.00%	5.44
22	48.28%	37.93%	44.83%	44.83%	34.48%	41.38%	48.28%	37.93%	44.83%	44.83%	42.76%	4.65
23	17.24%	20.69%	27.59%	24.14%	27.59%	37.93%	27.59%	31.03%	31.03%	34.48%	27.93%	6.18
24	37.93%	34.48%	41.38%	41.38%	37.93%	34.48%	37.93%	41.38%	31.03%	34.48%	37.24%	3.56
25	31.03%	34.48%	37.93%	31.03%	31.03%	41.38%	37.93%	48.28%	44.83%	34.48%	37.24%	6.04
26	44.83%	37.93%	34.48%	37.93%	34.48%	48.28%	17.24%	48.28%	41.38%	48.28%	39.31%	9.51
27	24.14%	37.93%	31.03%	31.03%	27.59%	27.59%	27.59%	20.69%	31.03%	31.03%	28.97%	4.65
28	41.38%	31.03%	41.38%	37.93%	27.59%	24.14%	55.17%	44.83%	34.48%	31.03%	36.90%	9.20
29	20.69%	20.69%	31.03%	44.83%	41.38%	27.59%	24.14%	31.03%	31.03%	44.83%	31.72%	9.17
30	48.28%	44.83%	48.28%	41.38%	44.83%	48.28%	37.93%	44.83%	48.28%	41.38%	44.83%	3.63
Mean	35.17%	36.78%	38.97%	37.59%	37.93%	38.97%	39.54%	38.85%	36.09%	37.93%		
Std. Dev.	9.69	8.07	7.58	7.38	7.84	7.37	10.31	9.28	7.51	8.92		