Binary Fish School Search applied to Feature Selection

João André Gonçalves Sargo

Thesis to obtain the Master of Science Degree in

Mechanical Engineering

Examination Committee

Chairperson: Prof. João Rogério Caldas Pinto
Supervisor: Prof. João Miguel da Costa Sousa
Members of the Committee: Prof. Jorge dos Santos Salvador Marques
Prof. Susana Margarida da Silva Vieira

November 2013
À minha mãe
Abstract

The aim of the present work is to develop efficient feature selection approaches. The problem regarding the increasingly larger accumulation of data is presented, where feature selection emerges as a promising solution. Despite the variety of feature selection methods, few of them are able to guarantee a good performance, especially in high dimensional databases.

A novel wrapper methodology applied to feature selection is formulated based on the Fish School Search (FSS) optimization algorithm, intended to cope with premature convergence. The FSS was originally designed with a real encoding scheme for searching high-dimensional spaces based in fish schools behaviour. In order to use this population based optimization algorithm in feature selection problems, the use of binary encoding for the internal mechanisms of the fish school search is proposed, emerging the binary fish school search (BFSS).

The proposed algorithm, as well as other state of the art feature selection methods such as Sequential Forward Selection (SFS) and Binary Particle Swarm Optimization (BPSO), were combined with fuzzy modelling in a wrapper approach and tested over two databases, a benchmark and an ICU (intensive care unit) database. The purpose of using this last database was to predict the readmission of ICU patients 24 to 72 hours after being discharged. Several statistical measures were considered to characterise the patient stay, including the Shannon entropy and the weighted mean.

The results obtained by comparing the performance measures and the number of features selected of the used algorithms, show promising results for the novel algorithm BFSS.

Keywords: Feature Selection, Binary Encoding, Fish School Search, ICU, Readmissions
Resumo

O presente trabalho visa o desenvolvimento de abordagens eficientes para o problema de seleção de variáveis. A questão da crescente quantidade de informação acumulada é debatida, para o qual a noção de seleção de variáveis se estabelece como uma solução promissor. Apesar da grande variedade disponível, poucos são os métodos capazes de garantir alta precisão, sobretudo em bases de dados de grande dimensão.

Neste sentido, uma nova metodologia foi aqui formulada com base no algoritmo de otimização *Fish School Search*, destinado a lidar com a convergência prematura das soluções. Este método, originalmente desenvolvido com um esquema de codificação em números reais, pesquisa espaços de alta dimensão baseando-se no comportamento de cardumes. De forma a utilizar este algoritmo de otimização em problemas de seleção de variáveis, foi proposto o uso de um esquema de codificação binária para os seus mecanismos internos, surgindo o *Binary Fish School Search* (BFSS).

O algoritmo aqui proposto, bem como outros métodos, *Sequential Forward Selection* e *Binary Particle Swarm Optimization*, foram conciliados com modelação *fuzzy* numa abordagem *wrapper* e testados em duas bases de dados, uma de *benchmark* e outra de uma unidade de cuidados intensivos (UCI). Esta última foi utilizada de modo a prever a readmissão de pacientes após alta. Foram consideradas várias medidas estatísticas para caracterizar a sua estadia, incluindo a entropia de Shannon e a média ponderada.

Os resultados obtidos, através da comparação da precisão e do número de variáveis selecionadas dos vários algoritmos usados, mostram resultados promissores para o novo algoritmo BFSS.

**Palavras-chave:** Seleção de Variáveis, Codificação binária, UCI, Readmissões
Acknowledgements

My first words of appreciation are to my supervisors Professor João Sousa and Dr. André Fialho, for all the help, support, and for the opportunity to work in this field of research. A special thanks to Professor Susana Vieira, for all the support and guidance when I needed the most. Her expert knowledge in the field of this work was a great contribution. I couldn’t forget to thank Professor Carmelo J. A. Bastos Filho and Débora N. de Oliveira Nascimento for the help and to their readiness in clarifying doubts.

I would like thank my colleagues at Health Care IST workgroup for their help, support and for the inspirational breaks.

To my parents, for all the love, hard work, dedication and for always being with me. A special thanks to my mother, her sacrifice made all this possible.

I would also like to thank to all the friends, in special to António Ramos, Magno Mendes and Renato Ribeiro for all the willingness, happiness and all the moments of true friendship. A special thanks Marta Ferreira, for her encouragement and inestimable support during all this work and until the very last moment.
Contents

Abstract ........................................................................................................................................... V
Resumo............................................................................................................................................... VII
Acknowledgements ........................................................................................................................... IX
List of Figures ....................................................................................................................................... XV
List of Tables ......................................................................................................................................... XVII
Notation ................................................................................................................................................ XIX

1 Introduction ........................................................................................................................................ 1
  1.1 Knowledge Discovery ................................................................................................................ 1
    1.1.1 Principles of Feature Selection .......................................................................................... 3
    1.1.2 Modeling ............................................................................................................................ 4
  1.2 Prediction of readmissions ......................................................................................................... 6
  1.3 Contributions ............................................................................................................................ 6
  1.4 Outline .......................................................................................................................................... 7

2 Knowledge Data Discovery ............................................................................................................. 9
  2.1 DATA ........................................................................................................................................... 9
    2.1.1 Benchmark database – Sonar ............................................................................................ 9
    2.1.2 MIMIC II database ........................................................................................................... 10
  2.2 Modeling ..................................................................................................................................... 16
    2.2.1 Fuzzy modeling ................................................................................................................. 16
    2.2.2 Clustering .......................................................................................................................... 17
    2.2.3 Performance Measures .................................................................................................... 18
  2.3 Feature Selection .......................................................................................................................... 20
    2.3.1 Sequential Forward Selection ......................................................................................... 20
    2.3.2 Binary Particle Swarm Optimization ............................................................................. 22

3 Fish school search ............................................................................................................................ 25
  3.1 Original Fish school search ........................................................................................................ 25
    3.1.1 Search Problems and Algorithms ..................................................................................... 26
3.1.2-FSS Computational Principles ................................................................. 26
3.1.3-Overview of the algorithm ........................................................................ 27
3.1.4-The Feeding Operator ................................................................................ 27
3.1.5-The Swimming Operators .......................................................................... 28
3.1.6-Individual Movement ................................................................................ 28
3.1.7-Collective-Instinctive Movement ............................................................... 29
3.1.8-Collective-Volitive Movement .................................................................. 30
3.1.9-FSS Cycle and Stop Conditions ................................................................. 31
3.1.10-Illustrative Example ................................................................................ 32
3.2- Decimal to binary Fish school search ......................................................... 36
3.2.1-Objective function .................................................................................... 36

4 Binary Fish School Search .................................................................................. 37
4.1- Encoding ........................................................................................................ 37
4.2-Initialization .................................................................................................... 38
4.3-Individual Movement ..................................................................................... 38
4.4-Collective-Instinctive Movement ................................................................ 38
4.5-Collective-volitive Movement ...................................................................... 40
4.5-Objective function ......................................................................................... 41
4.7-Parameters ..................................................................................................... 42
4.8- BFSS cycle and stop condition ................................................................... 42

5 Results .................................................................................................................. 43
5.1-Description of the approach .......................................................................... 43
5.2-Optimization Parameters .............................................................................. 44
5.3-Sonar database .............................................................................................. 46
5.3.1-Sequential Forward Selection ................................................................... 46
5.3.2 Decimal to Binary Fish School Search ...................................................... 47
5.3.3 Binary Fish School Search ........................................................................ 49
5.3.4 Binary Particle Swarm Optimization ......................................................... 53
5.3.5 Comparison of Feature Selection Methods .............................................. 54
5.4-Readmission database .................................................................................. 55
5.4.1 Sequential Forward Selection ................................................................... 56
5.4.2 Binary Fish School Search ........................................................................ 56
5.4.3 Binary Particle Swarm Optimization ......................................................... 59
5.4.4 No feature selection .................................................................................. 61

XII
List of Figures

1.1 Knowledge discovery process ........................................................................................................... 2
2.1 Patient selection flowchart .................................................................................................................. 12
2.2 Number of patients vs number of samples ......................................................................................... 14
2.3 Representation of the different gradients for the weighted mean .................................................. 15
2.4 Illustrative diagram of the inputs for the classification models ....................................................... 15
2.5 Example of ROC curve ....................................................................................................................... 19
2.6 Decoding process for D2BFSS ........................................................................................................ 22
3.1 FSS-Individual movement .................................................................................................................. 29
3.2 FSS-Collective-instinctive movement ............................................................................................... 30
3.3 FSS-Collective-volitive movement .................................................................................................. 31
3.4 FSS-example ...................................................................................................................................... 34
3.5 FSS-Fish school evolution ................................................................................................................ 35
4.1 Parameters of FSS, D2BFSS and BFSS ............................................................................................. 42
5.1 Diagram of the feature selection process ............................................................................................ 44
5.2 Evolution of SFS - sonar database .................................................................................................... 47
5.3 Evolution of B2DFSS - sonar database ............................................................................................. 49
5.4 Representation of the study of parameters - BFSS - sonar database .............................................. 50
5.5 Representation of the study of parameters - BFSS - sonar database - Continuation .................. 50
5.6 Representation of the study of parameters - BFSS - sonar database - Continuation .................. 51
5.7 Representation of the study of parameters - BFSS - sonar database - Continuation .................. 51
5.8 Representation of the study of parameters - BFSS - sonar database - Continuation .................. 51
5.9 Representation of the study of parameters - BFSS - sonar database - Continuation .................. 52
5.10 Evolution of BFSS - sonar database ................................................................................................ 53
5.11 Evolution of BPSO - sonar database ................................................................................................. 54
5.12 Representation of the study of parameters - BFSS - readmission database .................................. 57
5.13 Evolution of BFSS - readmission database ..................................................................................... 59
5.14 Evolution of BPSO - readmission database ..................................................................................... 61
List of Tables

2.1: List of physiological variables considered.................................................................11
2.2: Physiological limits.....................................................................................................13
2.3: Summary of the number of samples and patients......................................................14
2.4: Number of patients readmitted and not readmitted...................................................14
3.1: FSS - example..............................................................................................................33

5.1: Model assessment results - SFS - sonar database.......................................................46
5.2: Results of the study the parameters: step\textsubscript{end} and step\textsubscript{vol} - D2BFSS..................48
5.3: Results of the study the parameters: W\textsubscript{scale} and α - D2BFSS..........................48
5.4: Model assessment results - D2BFSS - sonar database...............................................49
5.5: Configuration of the parameters thres\textsubscript{c} and thres\textsubscript{v} tested - BFSS....................50
5.6: Results of the study the parameters: step\textsubscript{end} - BFSS........................................52
5.7: Model assessment results - BFSS - sonar database...................................................52
5.8: Model assessment results - BPSO - sonar database..................................................53
5.9: Comparison between FS algorithms - sonar database.............................................54
5.10: Model assessment results - SFS - readmission datasets........................................56
5.11: Configuration of the parameters thres\textsubscript{c} and thres\textsubscript{v} - BFSS..........................56
5.12: Results of the study the parameters: step\textsubscript{end} - BFSS........................................58
5.13: Model assessment results - BFSS - readmission datasets........................................58
5.14: Results of the study the parameters: α - BPSO..........................................................60
5.15: Model assessment results - BPSO - readmission datasets.......................................60
5.16: Model assessment results - NO FS - readmission datasets......................................61
5.17: Comparison between FS - readmission dataset......................................................62

A.1: Results of the study the parameters: thres\textsubscript{c} and thres\textsubscript{v} - Tests 1-3.........................ii
A.2: Results of the study the parameters: thres\textsubscript{c} and thres\textsubscript{v} - Tests 4-7..........................iii
A.3: Results of the study the parameters: thres\textsubscript{c} and thres\textsubscript{v} - Tests 8-11.......................iv
A.4: Results of the study the parameters: thres\textsubscript{c} and thres\textsubscript{v} - Tests 12-14......................v
A.5: Results of the study the parameters: thres\textsubscript{c} and thres\textsubscript{v} - Tests 15-17.....................vi
A.6: Results of the study the parameters: thres\textsubscript{c} and thres\textsubscript{v} - Test 18..........................vii
B.1: Results of the study the parameters: \textit{thres}_c and \textit{thres}_v - Tests 1-2 ............................................viii

B.2: Results of the study the parameters: \textit{thres}_c and \textit{thres}_v - Tests 3-5 ............................................ ix
Notation

Symbols

\( N_f \) - Number of features selected
\( N_t \) - Total number of features
\( N_s \) - Number of data samples
\( y \) - System output
\( x \) - Data sample
\( X \) - Database
\( \bar{y} \) - Model output
\( Y \) - Set of possible labels or classes
\( u \) - Class
\( l \) - Number of existing labels
\( D \) - Decision region
\( \tau \) - Threshold
\( w_y \) - Weight factor
\( \sigma_j \) - Activation function
\( A_i \) - Fuzzy set of the antecedent
\( B_i \) - Fuzzy set of the consequent
\( R_i \) - Fuzzy rule
\( R \) - Number of rules
\( \mu A(x), \mu B(y) \) - Membership functions
\( \mu R(x, y) \) - Fuzzy relation
\( f_i \) - Mapping function of the \( i \)th rule
\( \beta_i \) - Degree of fulfillment of rule \( i \)
\( v_i \) - Cluster center
\( J \) - Objective function
\( U \) - Fuzzy partition matrix
\( V \) - Matrix of cluster prototypes
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho_i$</td>
<td>Constant that controls volume of cluster $i$</td>
</tr>
<tr>
<td>$\pi_i$</td>
<td>Parameter vector for the $i$th rule</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Matrix of model parameters</td>
</tr>
<tr>
<td>$\psi$</td>
<td>Weighted vector of inputs</td>
</tr>
<tr>
<td>$\Psi$</td>
<td>Matrix of vector inputs</td>
</tr>
<tr>
<td>$P(x, y)$</td>
<td>Probability distribution function</td>
</tr>
<tr>
<td>$R[f]$</td>
<td>Risk or expected loss of $f$</td>
</tr>
<tr>
<td>$F$</td>
<td>Features space</td>
</tr>
<tr>
<td>$\varphi$</td>
<td>Non-linear mapping function</td>
</tr>
<tr>
<td>$x_0$</td>
<td>Subset of features</td>
</tr>
<tr>
<td>$\rho</td>
<td>x_0)$</td>
</tr>
<tr>
<td>$p$</td>
<td>Parameter value</td>
</tr>
<tr>
<td>$g_i$</td>
<td>Value of bit $i$</td>
</tr>
<tr>
<td>$u$</td>
<td>Number of bits used</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>Disturbance associated with real data</td>
</tr>
<tr>
<td>$O$</td>
<td>Objective</td>
</tr>
<tr>
<td>$C_{i(v)}$</td>
<td>Constraint function</td>
</tr>
<tr>
<td>$N_{e}$</td>
<td>Number of misclassifications</td>
</tr>
<tr>
<td>$N_{s}$</td>
<td>Total number of tested samples</td>
</tr>
<tr>
<td>$S(x_i)$</td>
<td>Cumulative relative fitness</td>
</tr>
<tr>
<td>$p_{tour}$</td>
<td>Tournament selection probability</td>
</tr>
<tr>
<td>$p_{c}$</td>
<td>Crossover probability</td>
</tr>
<tr>
<td>$p_{u}$</td>
<td>Uniform crossover probability</td>
</tr>
<tr>
<td>$p_{mut, r_{mut}}$</td>
<td>Mutation probability</td>
</tr>
<tr>
<td>$S$</td>
<td>Particle swarm</td>
</tr>
<tr>
<td>$V_i$</td>
<td>Visibility sphere of particle $i$</td>
</tr>
<tr>
<td>$v_i$</td>
<td>Particle velocity</td>
</tr>
<tr>
<td>$x_i$</td>
<td>Particle position</td>
</tr>
<tr>
<td>$a_i$</td>
<td>Particle acceleration</td>
</tr>
<tr>
<td>$\Delta t$</td>
<td>Time step</td>
</tr>
<tr>
<td>$o_i$</td>
<td>Particle cohesion</td>
</tr>
<tr>
<td>$l_i$</td>
<td>Particle alignment</td>
</tr>
<tr>
<td>$s_i$</td>
<td>Particle separation</td>
</tr>
<tr>
<td>$C_{o, c, c_s}$</td>
<td>Acceleration weighting factors</td>
</tr>
<tr>
<td>$S(v_{ij})$</td>
<td>Logistic function of the velocity</td>
</tr>
<tr>
<td>$v_{max}$</td>
<td>Particle velocity threshold</td>
</tr>
</tbody>
</table>
\( w_{up} \) - Upper bound weight
\( w_{low} \) - Lower bound weight
\( \text{Stepind} \) - step individual parameter
\( \text{Stepvol} \) - step volative parameter
\( t \) - iteration
\( \text{gactual} \) - actual iteration
\( g_{total} \) - total number of iterations
\( \text{thres}_c \) - collective threshold parameter
\( \text{thres}_v \) - volative threshold parameter

**Acronyms**

ACC - Accuracy
AUC - Area Under the Curve
BFSS - Binary Fish School Search
BPSO - Binary Particle Swarm Optimization
D2BFSS - Decimal to Binary Fish School Search
FCM - Fuzzy C-Means
FM - Fuzzy Modelling
FN - False Negatives
FP - False Positives
FS - Feature Selection
FSS - Fish School Search
GA - Genetic Algorithm
ICU - Intensive Care Unit
IOM - Institute of Medicine
IQR - Interquartile Range
KDD - Knowledge Data Discovery
MA - Model Assessment
NAE - National Academy of Engineering
NN - Neural Networks
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP-hard</td>
<td>Non-deterministic Polynomial-time hard</td>
</tr>
<tr>
<td>PSO</td>
<td>Particle Swarm Optimization</td>
</tr>
<tr>
<td>ROC</td>
<td>Receiver Operating Characteristic</td>
</tr>
<tr>
<td>SA</td>
<td>Simulated Annealing</td>
</tr>
<tr>
<td>SFS</td>
<td>Sequential Forward Selection</td>
</tr>
<tr>
<td>TN</td>
<td>True Negatives</td>
</tr>
<tr>
<td>TS</td>
<td>Takagi-Sugeno</td>
</tr>
<tr>
<td>TP</td>
<td>True Positives</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Over the past 30 years, the continuing development and application of systems engineering methods has enabled an unprecedented growth in the manufacturing, logistics, distribution, and transportation sectors of our economy [35]. Vast business organizations (e.g. airline companies, chains of department stores, large manufacturing companies), could not properly operate in the current business environment without the extensive use of various engineering tools for design, analysis and control of complex production and distribution systems [36]. However, even with the emergence and development of new engineering techniques over the past years, some industries have barely begun to take advantage of systems engineering tools, which means that there is a great number of potential unexplored applications for them.

A good example is the health care delivery system, one of the most technologically intense and data-rich industries [32]. According to a report from the National Academy of Engineering (NAE) and the Institute of Medicine (IOM), the application of systems engineering tools could play a crucial role in solving the current crisis in the very complex health care system [47].

With the computerization of many sectors and with the advances in data collection tools, our capabilities of both generating and collecting data have been increasing rapidly in the last several decades. This explosive growth in stored data has generated an urgent need for new techniques and automated tools that can intelligently assist us in transforming the vast amounts of data into useful information and knowledge [26].

This chapter begins with a brief overview of methods currently used in knowledge discovery in databases. Further, the case of study will be introduced, the prediction of readmissions in an intensive care unit (ICU). In the end of the chapter, the contributions and outline of this work are presented.

1.1-Knowledge Discovery

The information age is very hard to grasp. In an average person’s life nowadays, we get more information in a day than someone who lived 100 years ago would get in a lifetime. The speed at
which information is increasing means that finding accurate data is becoming more important than the data itself [42].

The traditional method of turning data into knowledge relies on manual analysis and interpretation. In the health care industry, this form of manual probing of a data set is slow, expensive and highly subjective. With the urgent need for a new generation of computation techniques and tools to assist humans in extracting useful information (knowledge) from a fast growing volume of data, a methodology was created, the Knowledge Data Discovery (KDD), first introduced by Fayyad in 1996 [14].

The KDD process can be formally defined as a non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in large amounts of data.

![Figure 1.1: Knowledge discovery process.](image)

The KDD process can be decomposed into five main steps, as illustrated in Fig. 1.1:

1. **Data acquisition** - The process of acquiring and storing data.
2. **Data Preprocessing** - Consists of applying proper techniques that allow the improvement of the overall quality of the data. Includes processing of noise/outliers, correction of missing values, and/or alignment of data sampled at different frequencies.
3. **Feature Selection** - Consists of finding useful features (variables) to represent the data and discarding the non-relevant ones, containing redundant information.
4. **Modeling** - Refers to the process of combining methods from computational intelligence and/or statistics to extract patterns in data sets. In this work it was used classification models, which identifying to which of a set of categories (classes) a new observation belongs, on the basis of a training set of data containing observations whose category membership is known.
5. **Interpretation** - The process of evaluating the discovered knowledge with respect to its validity, usefulness, novelty, and simplicity. External expertise may be required in this step.

All of the five steps described are equally crucial, and the process is iterative, i.e. multiple loops can occur between any steps of the KDD method.

In real-world systems, the selection of a low number of features that consistently describe the problem is usually time consuming and, in many cases, impossible to achieve with a greedy approach. In this work, the focus is turned to the feature selection stage of the KDD method. A novel approach is
proposed for the optimization algorithm Fish School Search (FSS), in order to use this population based algorithm in problems of feature selection.

1.1.1-Principles of Feature Selection

The addition of more features (variables) is expected to increase the accuracy of the model (classifier). However, for some classifiers an increase in input dimensionality decreases the reliability of statistical parameter estimations and may, consequently, result in a decrease in the classification accuracy [43]. This is known as the Hughes effect [29], the so-called curse of dimensionality, which postulates that the classification accuracy will decrease after a certain feature-set size is reached unless the number of training samples is proportionally increased [43]. The Hughes effect is therefore more likely to be encountered when small training sets are used and the input dimensionality is increased.

The field of feature selection has been object of extensive research in recent years [41]. This is explained due to the potential benefits introduced when reducing data dimensionality. It can greatly improve data visualization and understanding, facilitating knowledge discovery. Furthermore, one needs to measure and store less information leading to a reduction in equipment, and consequently cutting unnecessary costs. From the clinical point of view, this process may bring to light new variables that had not been previously considered as relevant for a given medical problem.

Feature selection algorithms can be grouped into four categories: filters, wrappers, hybrids and embedded [52, 23]. Filter methods rely on general characteristics of the data to evaluate and select feature subsets without involving any mining algorithm. Some examples include using measurements of entropy, variance, correlation or mutual information of single and multiple variables [53]. Wrappers require one predetermined mining algorithm and use its performance as the evaluation criterion. They search for the features best suit to improve the performance of the mining algorithms, but they also tend to be more computationally expensive than filters [56]. Some of the most commonly used wrapper methods include best-first, branch-and-bound, simulated annealing, genetic algorithms, forward selection or backward elimination, but the list is considerably longer and continuously growing. The present thesis introduces a new wrapper method, the Binary Fish School Search (BFSS) algorithm.

Hybrid models attempt to take advantage of the two previous types of models by exploiting their advantages in different stages [22]. First, a filter decreases the dimensionality of data by eliminating features according to the specified criteria. Then, wrappers select relevant features according to the mining objective.

Finally, embedded methods differ from the previous feature selection methods in the way feature selection and learning interact. In contrast to filter and wrapper approaches, the learning and
feature selection parts cannot be separated in embedded methods [23]. Examples of embedded methods for feature selection include decision trees and random multinomial.

Many problems related to Feature Selection (FS) have been shown to be NP-hard and finding the optimal set of features is usually intractable [6,30]. Thus, the search of the most predictive feature subsets can be seen as an optimization problem. Metaheuristics are general upper level (meta) algorithmic techniques, that can be used as guiding strategies in the design of heuristics to solve specific optimization problems. These techniques are capable of finding acceptable solutions, within reasonable time, by using experience-based techniques or through guided search, but do not guarantee that the optimum will be found. Popular metaheuristics for combinatorial problems include simulated annealing (SA) by Kirkpatrick [34] genetic algorithms (GA) [27] Scatter Search [19], Tabu Search [20], and Particle Swarm optimization (PSO).

The present thesis resorted to: 1) a new FS wrapper method based on the new-found metaheuristic Fish school Search optimisation, based on fish school behaviour [17], 2) a PSO algorithm modified to be used in FS problems [15] and 3) a wrapper method based on Tree search feature selection: the Sequential Forward Selection (SFS). The three algorithms were tested in a benchmark database before being applied to a problem in the health care system.

1.1.2- Modeling

Classification modeling, used in the data mining process, can be defined as the application of discovery algorithms that produce a particular enumeration of patterns/models over the data.

The usefulness of a model is to mimic how a particular object or phenomenon will behave in a particular condition. It can be used for testing, analysis or training, in conditions where real-world systems or concepts can be represented by a model [54].

Machine learning refers to a group of mathematical modeling techniques that are capable of automatically acquiring and integrating knowledge based on empirical data, such as data from sensors or databases. This area has been extensively studied with numerous successful applications across a wide range of fields (a very broad description of application areas and examples can be found in [33]).

The purpose of using machine learning techniques (or learning machines) is to reproduce the human learning capabilities, namely the ability to recognize complex patterns and make intelligent decisions based on data.

Learning machines are widely used in classification, regression, recognition and prediction problems. There are many possible applications for these modeling techniques, that range from engineering applications in robotics, fault tolerant control, pattern recognition (e.g. speech recognition, handwriting recognition), to medical applications (e.g. diagnosis, prognosis) [33].
Nonetheless, in this work, the main interest is the machine learning capability of discovering and classifying patterns in high dimension databases.

Pattern recognition [11, 44, 46] addresses the problem of assigning labels (classes) to objects (or samples), being each sample composed by a set of features (or attributes). In order to better understand pattern recognition, this subject has been divided into two major types of problems [37]: unsupervised and supervised learning.

In the unsupervised category, the problem is to understand whether there are groups in the data, and what characteristics make the objects similar within the group and different across the groups. Contrarily, in supervised learning, each data sample already has a pre-assigned label, and the task consists of training a classifier in order to differentiate between labels.

In this work, it was decided to use a non-linear machine learning technique, Fuzzy Modelling (FM). This method is considered suitable for the demanding problem of pattern recognition, since it can, theoretically, approximate any multivariate nonlinear function [37]. The main advantages of this method are the following:

- Efficient tool for embedding human (structured) knowledge into useful algorithms multivariate;
- Applicable when mathematical model is unknown or impossible to obtain;
- Operates successfully under a lack of precise sensor information;
- Useful at the higher levels of hierarchical control systems;
- Appropriate tool in generic decision-making process;
- Transparent, non-crisp model;
- Interpretation in the form of rules and logical connectedness. From the medical point of view, these rules provide additional means of validating the fuzzy classifier by clinician’s knowledge regarding the system.

The main disadvantages are:

- Experts may have problems in structuring the knowledge with respect to the structure of the model;
- Experts sway between extreme poles: too much aware in field of expertise, or tending to hide their knowledge;
- Model complexity increases exponentially with the increase of the number of features;
- Learning is highly constrained; typically more complex than other models, like neural networks (NN).
1.2 Prediction of readmissions

Patients readmitted to an intensive care unit during the same hospitalization have an increased length of stay, higher costs and increased risk of death. Previous studies have demonstrated overall readmission rates of 4-14% [3, 48], of which nearly a third can be attributed to premature discharge from the critical care setting [3, 12]. It is also documented that the length of stay for readmitted patients is at least twice as long as that for patients discharged from the ICU but not readmitted and that hospital death rates are 1.5 to almost 10 times higher among ICU readmission [49].

Increasing pressures on managing care and resources in ICUs is one explanation for strategies seeking to rapidly free ICU beds. Faced with this scenario, a clinician may elect to discharge a patient, currently in the ICU, who has already had the benefits of stabilization and intensive monitoring, to make room for more acute patients allocated in the emergency department, exposing the outwardly transferring patients to the risk of readmission in the short term. Moreover, despite the existence of morbidity and mortality issues around readmission, the Centres for Medicare & Medicaid Services have already reduced funding for specified avoidable conditions, and it is quite possible that avoidable readmission to an ICU will receive attention in the future as well.

Previous studies [7] have examined different variables that are assessed at discharged, but these predictive models performed only slightly better than models based upon the gold standard method- APACHE II.

Thus, this work has encountered the problem of readmission to an ICU, being its goal to predict the readmission of patients in an ICU within 24-72 hours after the discharge. A data mining approach was used to a real world database, the MIMIC II, combined with fuzzy modelling and three different Feature Selection algorithms, the Sequential Forward Selection, the Binary Particle Swarm optimization and the novel Binary Fish School Search, here formulated. In this context, 22 physiologic variables acquired during the stay of real patient in an ICU were selected. Statistical measures were utilized to describe each patient stay: the mean, the standard deviation, the maximum, the minimum, the Shannon entropy and the weighted mean, which was tested for different weights.

1.3 Contributions

In this work, the problem of Feature Selection in real-world databases is addressed. The main contributions of this work are:

- Introduction and formulation of Binary Fish School algorithm, the novel algorithm for Feature Selection derived from the optimization algorithm Fish School Search. Originally this algorithm was presented as a multidimensional real system encoded algorithm [17], and is here modified in
order to solve problems with binary inputs. The algorithm was then applied to feature selection problems;

- Use of new types of features (weighted mean and Shannon entropy) to predict the readmissions of patients in the ICU during the 24 to 72h period that follows the discharge;
- Comparison of the FS results applied to two real databases using the three feature selection algorithms: sequential forward selection, particle swarm optimization and binary fish school algorithm.

1.4 Outline

In chapter 2, an overview of the knowledge data discovery stages, studied in this work, is presented. It begins with the definition of the two addressed databases and the necessary preprocessing of the data. Then, the fuzzy modelling technique is presented, together with the performance measures considered in this work. Finally a broad description of wrapper methods is presented along with description of the state of the art feature selection algorithms used in this work.

In chapter 3, the original fish school search algorithm is presented in detail. The internal mechanisms are featured and an illustrative example is given to consolidate the description. Finally the first approach to transform the FSS algorithm in order to solve feature selection problems is presented, the decimal to binary fish school search.

Chapter 4 will introduce the goals and detailed formulation of the binary fish school search.

Chapter 5 presents the results of the wrapper methods that combine the studied machine learning techniques with the introduced search algorithms. The chapter begins with the presentation of the outline of the approach, and the definition of the parameters used to evaluate the functioning of the formulated algorithms. Next, the tests to select the parameters for the optimization algorithms are presented. Finally, the methods are tested and compared over the two databases.

At last, in Chapter 6 the results of this work are summarized and conclusions are drawn. Furthermore, promising areas for future research are presented.
Chapter 2

Knowledge Data Discovery

2.1-DATA

In this work, two databases were used: a benchmark database and a health care database, the MIMIC II. The benchmark databases are employed to ascertain the quality of the developed FS algorithms, i.e., to verify if the FS algorithms are capable of selecting a low number of features subset with good informative potential. After validation with the benchmark databases, the FS algorithms were applied to a health care database, the MIMIC II, as a prediction of readmission problem.

In this chapter, the selected benchmark database is initially exposed and then the health care database is presented as well as the necessary processing for this database.

2.1.1-Benchmark database – Sonar

The choice of a proper group of benchmark databases is very important to adequately validate the implementation of an algorithm. These databases should allow the algorithm designer to test the algorithms according to the predefined performance measures and allow the comparison between these results with those from state of the art methods.

The sonar database is comprised of 208 real samples of rocks that are divided in two labels. A data sample is a set of 60 features with values ranging from 0.0 to 1. Each of these features represents the energy within a particular frequency band, integrated over a certain period of time. The label associated with each record contains the indication if the rock sonar signals bounced off a metal cylinder (97 samples) or bounced off a roughly cylindrical rock (111 samples). The task at hand was to discriminate between these two classes [21].

This database was developed by Sejnowski and Gorman on their study in the classification of sonar signals using artificial neural networks [21].
2.1.2- MIMIC II database

The MIMIC II database [51] is a large database of ICU patients admitted to the Beth Israel Deaconess Medical Centre, collected from 2001 to 2006. The MIMIC II database is currently formed by 25,549 patients, of which 19,075 are adults (> 15 years old at time of admission). For each patient, several samples of physiological variables were stored throughout their stay.

In this work, a previously developed dataset (first presented in [15]) was used, including only adult patients (>15 years) that were ICU inpatients for at least 24 h and readmitted back to any ICU of the same medical centre between 24 and 72 h. This interval is often referred to as an early readmission [43]. The reason for choosing 24 h as the lower bound for the readmission time window is related to how MIMIC II is structured. Also, patients readmitted to the ICU less than 24 h after their discharge are considered to belong to the same ICU stay. The choice of 72 h as the upper bound for the readmission time window was based on previous works [50], and local clinical intensivist suggestions. All included patients were also required to have at least one measurement of the 22 variables shown in Table 2.1. These variables were selected based on the hypothesis that a good predictive value could be achieved using a few physiological variables and taking into account the following directives:

i. The variables had to be easily and/or routinely assessed in the 24 h before discharge. A balance had to exist in the number of selected variables given that it will affect the number of patients that will form the dataset, i.e. the more variables defined, the fewer the patients that were likely to have all of them collected at the same time;

ii. Selecting a high number of variables may bias the dataset towards selecting patients having similar conditions that required their specific measurement/testing;

iii. The variables chosen should be independent with minimal correlation.

Exclusion criteria included patients who died during the ICU stay.

As with other real-world databases, a few preprocessing steps were necessary to improve the quality of the raw data of the MIMIC II. In order to deal with variables collected within different sampling periods, similarly to [15], a template variable was used. This process aligned all samples to the same point in time as a designated template variable. Heart rate was chosen as the template variable on the basis since it was one of the most frequently measured variables and thus, introduced fewer artifacts in the data. With regards to missing data, in general, ICU data can be missing either because they are perceived to be irrelevant for the current clinical problems (thus, not recorded), or because exogenous interventions or endogenous activities have rendered the data useless [58].
Table 2.1: List of physiological variables considered from MIMIC II, (according to [15]).

<table>
<thead>
<tr>
<th>Type of variables</th>
<th>Variable name (units)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monitoring siginals</td>
<td>Heart rate (beats/min)</td>
</tr>
<tr>
<td></td>
<td>Respiratory rate (breaths/min)</td>
</tr>
<tr>
<td></td>
<td>Temperature(ºC)</td>
</tr>
<tr>
<td></td>
<td>SpO2(%)</td>
</tr>
<tr>
<td></td>
<td>Non-invasive arterial Blood pressure (systolic) (mmHg)</td>
</tr>
<tr>
<td></td>
<td>Blood pressure (mean) (mmHg)</td>
</tr>
<tr>
<td>Laboratory tests</td>
<td>Red blood cell count (cells x 10^3 /L)</td>
</tr>
<tr>
<td></td>
<td>White blood cell count (cells x 10^3 /L)</td>
</tr>
<tr>
<td></td>
<td>Platelets (cells x 10^3 /L)</td>
</tr>
<tr>
<td></td>
<td>Hematocrit (%)</td>
</tr>
<tr>
<td></td>
<td>BUN (mg/dL)</td>
</tr>
<tr>
<td></td>
<td>Sodium (mg/dL)</td>
</tr>
<tr>
<td></td>
<td>Potassium (mg/dL)</td>
</tr>
<tr>
<td></td>
<td>Calcium (mg/dL)</td>
</tr>
<tr>
<td></td>
<td>Chloride (mg/dL)</td>
</tr>
<tr>
<td></td>
<td>Creatinine (mg/dL)</td>
</tr>
<tr>
<td></td>
<td>Magnesium (mg/dL)</td>
</tr>
<tr>
<td></td>
<td>Albumin (g/dL)</td>
</tr>
<tr>
<td></td>
<td>Arterial pH</td>
</tr>
<tr>
<td></td>
<td>Arterial base excess (mEq/L)</td>
</tr>
<tr>
<td></td>
<td>Lactic acid (mg/dL)</td>
</tr>
<tr>
<td>Other</td>
<td>Urine output (mL/h)</td>
</tr>
</tbody>
</table>

Data missing for an intentional reason (e.g. patient is transported out of the ICU for an imaging scan) was considered non-recoverable and thus deleted. On the other hand, data missing for some unintentional reason (e.g. sensor goes off patient’s chest) was considered recoverable and the last available value was used to impute values to these segments.

The Interquartile Range (IQR) method was used in order to deal with the outliers. This method measures the statistical dispersion of the data, and divides it into quartiles. IQR is a trimmed estimator that identifies the most robust measure of scale [58]. The patient selection process is summarized in Fig. 2.1.

It is important to point out that the number of samples for each patient is not constant. A sample contains measures of the 22 physiologic variables. The number of these samples acquired for each patient during his stay can vary between 1 and 26 samples and it can have different sampling periods. The total number of samples considered was 13675.

It was detected that some samples of the 1028 selected patients contained outliers, so some preprocessing was necessary.

In order to use a constant dimension for the inputs of the models (necessary condition) a transformation to the data was performed. Statistical measures were used in order to seize the information of the time series for the physiologic variables of each patient.

The next section exposes the preprocessing used on this dataset.
Data preprocessing

Outliers

After analysing all the 13675 samples of the 1028 patients, and although the IQR method was applied to the raw database of MIMIC II, there were still some samples that contained values out of the physiologic limits. As an example, Fig. 2.2 shows the plot of the physiologic variable temperature (ºC) for all samples considered. The corporal temperature of 5 ºC seen in Fig. 2.2 is not possible, even if the patient is in a severe condition.

Figure 2.2: Graphical representation of the physiological variable temperature ºC for all samples.
The outliers were eliminated using the maximum and minimum limits for the 22 physiological variables of Table 2.2. These physiologic limits were obtained through the Decreased Variable Analysis (MEDAN).

Table 2.2: Physiological limits considered for the exclusion of outliers.

<table>
<thead>
<tr>
<th>No.</th>
<th>Variable name (units)</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Heart rate (beats/min)</td>
<td>0</td>
<td>250</td>
</tr>
<tr>
<td>2</td>
<td>Respiratory rate (breaths/min)</td>
<td>0</td>
<td>200</td>
</tr>
<tr>
<td>3</td>
<td>Temperature(ºC)</td>
<td>25</td>
<td>42</td>
</tr>
<tr>
<td>4</td>
<td>SpO2(%)</td>
<td>60</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>Non-invasive arterial Blood pressure (systolic) (mmHg)</td>
<td>30</td>
<td>300</td>
</tr>
<tr>
<td>6</td>
<td>Blood pressure (mean) (mmHg)</td>
<td>10</td>
<td>187</td>
</tr>
<tr>
<td>7</td>
<td>Red blood cell count (cells x 103/lL)</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>White blood cell count (cells x 103/lL)</td>
<td>0.4</td>
<td>50</td>
</tr>
<tr>
<td>9</td>
<td>Platelets (cells x 103/lL)</td>
<td>3</td>
<td>1000</td>
</tr>
<tr>
<td>10</td>
<td>Hematocrit (%)</td>
<td>19</td>
<td>60</td>
</tr>
<tr>
<td>11</td>
<td>BUN (mg/dL)</td>
<td>4</td>
<td>500</td>
</tr>
<tr>
<td>12</td>
<td>Sodium (mg/dL)</td>
<td>120</td>
<td>160</td>
</tr>
<tr>
<td>13</td>
<td>Potassium (mg/dL)</td>
<td>2.2</td>
<td>8</td>
</tr>
<tr>
<td>14</td>
<td>Calcium (mg/dL)</td>
<td>7.2</td>
<td>12</td>
</tr>
<tr>
<td>15</td>
<td>Chloride (mg/dL)</td>
<td>80</td>
<td>130</td>
</tr>
<tr>
<td>16</td>
<td>Creatinine (mg/dL)</td>
<td>0.1</td>
<td>9</td>
</tr>
<tr>
<td>17</td>
<td>Magnesium (mg/dL)</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>18</td>
<td>Albumin (g/dL)</td>
<td>0.5</td>
<td>18</td>
</tr>
<tr>
<td>19</td>
<td>Arterial pH</td>
<td>4.8</td>
<td>7.8</td>
</tr>
<tr>
<td>20</td>
<td>Arterial base excess (mEq/L)</td>
<td>-30</td>
<td>20</td>
</tr>
<tr>
<td>21</td>
<td>Lactic acid (mg/dL)</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>22</td>
<td>Urine output (mL/h)</td>
<td>0</td>
<td>1000</td>
</tr>
</tbody>
</table>

All of the samples that contain one or more physiologic variables with values out of the limits of Table 2.2 were considered samples with outliers. The total number of measures considered as outliers was 517. However, only 473 samples contain one or more variables with values out of the limits. According to [8] the missing samples, considered outliers, can be treated in various ways:

1. Ignore the tuple.
2. Fill in the missing value manually.
3. Use a global constant to fill in the missing value.
4. Use the attribute mean to fill in the missing value.
5. Use the attribute mean for all samples belonging to the same class as the given tuple.
6. Use the most probable value to fill in the missing value.

Methods 3 to 6 bias the data.

Since the number of samples, as well as the temporal spacing, of each sample for each patient are very irregular, models created later must have the ability to handle these irregularities. Thus, the approach 1 was chosen, in which a sample containing one or more measures outside of the limits in Fig. 2.2 is removed, being a process that does not bias data.
In this process, some patients had all their samples removed, resulting in a total of 1010 patients. Table 2.3 summarizes the preprocessing of the outliers.

Table 2.3: Summary of the number of samples and patients, resulting from preprocessing.

<table>
<thead>
<tr>
<th></th>
<th>No. of samples</th>
<th>No. of patients</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before the preprocessing</td>
<td>13675</td>
<td>1028</td>
</tr>
<tr>
<td>Removed during the preprocessing</td>
<td>473</td>
<td>18</td>
</tr>
<tr>
<td>After the preprocessing</td>
<td>13202</td>
<td>1010</td>
</tr>
</tbody>
</table>

The number of samples of the 22 physiologic variables per patient after treatment of outliers was analyzed, Fig.2.2 shows the variation of the number of patients per number of samples after the outlier’s treatment. It is worth noticing that the number of patients with only one sample is quite considerable.

![Figure 2.2: Graphical representation of the number of patients per number of measurements of the 22 physiological variables considered.](image)

In order to evaluate if the percentage of readmitted patients remained in the 4-14%, referred in the literature, an analysis of the percentage of patients readmitted was made, varying the patients with a minimal number of measurements for the 22 physiologic variables. Table 2.4 shows the results of this analysis for the variation of a minimum number of samples of 1 up to 10.

Table 2.4: Analysis of the number of patients readmitted and not readmitted for the subset of patients with a minimum number of samples. The percentage of readmitted patients remain in 4-14% as referred in the literature.

<table>
<thead>
<tr>
<th>Minimum number of samples:</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of patients not readmitted:</td>
<td>879</td>
<td>655</td>
<td>637</td>
<td>631</td>
<td>617</td>
<td>604</td>
<td>588</td>
<td>578</td>
<td>567</td>
<td>551</td>
</tr>
<tr>
<td>No. of patients readmitted:</td>
<td>131</td>
<td>94</td>
<td>89</td>
<td>89</td>
<td>88</td>
<td>87</td>
<td>86</td>
<td>82</td>
<td>80</td>
<td>78</td>
</tr>
<tr>
<td>% readmitted:</td>
<td>13.0</td>
<td>12.6</td>
<td>12.3</td>
<td>12.4</td>
<td>12.5</td>
<td>12.6</td>
<td>12.8</td>
<td>12.4</td>
<td>12.4</td>
<td>12.4</td>
</tr>
</tbody>
</table>

Data transformation

Knowing that there was a great variability in the number of samples per patient and very irregular sample periods, some descriptive statistics measures were used to describe the stay of each patient. By doing this, all patients would have the same number of features that described the time series of the physiologic variables throughout their ICU hospitalization. These features, with constant dimension, could then be used as inputs for the classification models.
Previous studies [15], used the arithmetic mean, the maximum, the minimum and the standard deviation of each physiologic variables in order to absorb the information of the time series of the considered physiologic variables for each patient. In the present work, in addition to these statistical measures, the Shannon entropy and the weighted average were also used, giving the possibility to withdraw more information.

Shannon entropy is the average unpredictability in a random variable, which is equivalent to its information content. It provides an absolute limit on the best possible lossless encoding or compression of any communication, assuming that the communication may be represented as a sequence of independent and identically distributed random variables. There are already studies that use entropy as feature extraction measure [45, 10].

In relation to the weights of the weighted mean, a linear distribution along the stay of the patient was considered, giving more relevance to the last measurements before the discharge. Four gradients were considered for these weights, as presented in Fig. 2.3.

In order to use the descriptive statistics measures announced before, it was decided to use only patients with a minimum of 3 measurements available, considering 725 patients (647 not readmitted and 89 readmitted, see Table 2.4). Thus, after the treatment of outliers and transformation of the dataset, 4 datasets emerged, one for each gradient of the weighted mean.

The only features that differ in each dataset are the 22 features that correspond to the weighted mean. Each dataset was formed by 726 patients (12.3% readmitted) and 132 features. The four datasets considered will be referred as readmition datasets.

Each patient will be considered as a sample for inputs of the classification models, Fig. 2.4 illustrates a sample.

Figure 2.3: Graphical representation of the four different gradients for the weights to be used in the weighted mean. The measures of the 22 variables vary between 0 and 24 hours before the discharge.

Figure 2.4: Illustrative diagram of the formation of the inputs for the classification models: each patient represents a sample ([1x132] array). Associated with each patient there is also a notification that indicates whether he is or not in readmitted class.
2.2-Modeling

In the present work, we used the machine learning technique of fuzzy modelling. These models were used as classification models. Briefly, for a given sample the model, created based on the train set of the data, is supposed to correctly assign this sample to one of the labels considered in the problem. An overview of the fuzzy modelling is given in the next topics.

2.2.1 Fuzzy modeling

Fuzzy modeling is a tool that allows approximation of nonlinear systems when there is little or no previous knowledge of the problem to be modeled [55, 13]. This tool supports the development of models around human reasoning (also referred to as approximate reasoning), and allows an element to belong to a set to a degree, indicating the certainty (or uncertainty) of its membership.

Within medical-related classification problems, several fuzzy-based models have shown comparable performances to other nonlinear modeling techniques [18, 16, 28]. Fuzzy modelling is particularly appealing as it provides not only a transparent, non-crisp model, but also a linguistic interpretation in the form of rules and logical connectives. These are used to establish relations between the defined features in order to derive a model. A fuzzy classifier contains a rule base consisting of a set of fuzzy if-then rules together with a fuzzy inference mechanism. These systems ultimately classify each instance of a dataset as pertaining to one of the possible classes defined for the specific problem being modeled [55].

For both databases used in this work (Sonar and readmission), the goal was to classify the samples in one of two labels. In the case of sonar database: rock sonar signals bounced off a metal cylinder or bounced off a roughly cylindrical rock, and in the readmission problem, patient would be readmitted or patient would not be readmitted to the ICU after 24-72 hours of discharge.

First order Takagi-Sugeno (TS) fuzzy models [55] were applied, which consist of fuzzy rules where each rule describes a local input-output relation. When first order TS fuzzy systems are used, each discriminant function consists of rules of the type:

\[ R_j: \text{If } x_1 \text{ is } A_{i1} \text{ and } \ldots \text{ and } x_N \text{ is } A_{iN} \text{ then } y_j = (a_j)^T x + b_j \]

where, \( j = 1, \ldots, J \) corresponds to the rule number, \( x = (x_1, \ldots, x_N) \) is the input vector, \( N \) is the total number of inputs (features), \( A_{in} \) is the fuzzy set for rule \( R_j \) and \( n^{th} \) feature, and \( y_j \) is the consequent function for rule \( R_j \).

The degree of activation of the \( j^{th} \) rule is given by:

\[ \beta_j = \prod_{n=1}^{N} \mu_{A_{jn}}(x) \]  
(2.1)
where $\mu_{A_j}(x) : R \rightarrow [0, 1]$.

The overall output is determined through the weighted average of the individual rule outputs. The number of rules, and the antecedent fuzzy sets $A_{j_n}$ are determined using fuzzy clustering in the product space of the input and output variables [55]. The consequent parameters for each rule $j$, are obtained as a weighted ordinary least-square estimate.

Given a classification problem, and being a linear consequent, a threshold $t$ is required to turn the continuous output $y \in [0, 1]$ into the binary output $y \in \{0, 1\}$. In this way, if $y < t$ then $y = 0$, and if $y \geq t$ then $y = 1$.

The number of rules $j$ and the antecedent fuzzy sets $A_{j_n}$ were determined by means of fuzzy clustering in the product space of the input and output variables.

For each database, the data was divided into training and test sets, while the model parameters were calculated using the training set, the feature subset quality was assessed using the test samples. Such approach was necessary due to the risk of overfitting, which means that the model could describe random error or noise instead of the underlying information. Thus, the test set provided a fair comparison over the generalization capabilities of the evaluated models [47].

### 2.2.2 Clustering

Clustering is an unsupervised learning method that organizes and categorizes data based on the similarity of data objects [2]. It is used in various fields, such as pattern recognition, machine learning and bioinformatics [33]. It is useful for knowledge discovery from empirical data and model construction.

A cluster can be seen as a group of objects more similar to one another than to other data points, being similarity usually defined as a distance norm. Furthermore, a cluster $i$ can also be seen as the area of influence of rule $R_i$. Therefore, a cluster center, also called prototype, coincides with the corresponding rule centre. The closer a data point is to a cluster center, the higher the fulfilment degree will be.

There is a great number of clustering algorithms, however, most of the analytical clustering algorithms are based on the minimization of the fuzzy c-means objective functional [5]. This objective function can be written as (2.2).

\[
J(X, U, V) = \sum_{i=1}^e \sum_{j=1}^{N_i} \mu_{ij}^m D_{ijA}^2
\]  

(2.2)

where the positive constant $m \in [1, \infty]$ determines fuzziness of the resulting clusters. The vector $x_j$ is one of the $N$ data samples, $v_i$ is the $i^{th}$ cluster center and $D_{ijA}$ is a distance norm between data points and cluster centers. The fuzzy partition matrix $U$ contains all the normalized membership values $\mu_{ij}$, $X$ is the matrix containing all the data samples, and $V$ is the matrix of the cluster prototypes.
In this work, the fuzzy C-means (FCM) clustering algorithm was used, requiring the definition of the number of cluster (which translates into the number of fuzzy rules). The number of clusters to be used was determined based on the minimization of the partition index [4]. This index accounts for both properties of the fuzzy memberships and structure of the data by measuring the compactness and separation of the clusters. This index is defined as:

\[ SC(J) = \sum_{j=1}^{l} \frac{\sum_{n=1}^{N} (\mu_{jn})^m \|x_n - v_j\|^2}{\sum_{q=1}^{l} \|v_q - v_j\|^2} \]  

(2.3)

where \( m \) corresponds to the weighting exponent of the FCM algorithm, \( x_j \) corresponds to the cardinality of fuzzy cluster \( j \), \( \|x_n - v_j\|^2 \) corresponds to the distance between a data point \( x_n \) and its cluster center \( v_j \), and \( \sum_{q=1}^{l} \|v_q - v_j\|^2 \) (named as the separation \( s_j \) of a fuzzy cluster \( j \)) corresponds to the sum of the distances from the cluster center \( v_j \) to the centers of all other \( J - 1 \) clusters. The lower the value of \( SC(J) \) the more compacted and separated are the clusters.

For the sonar databases and the four datasets of the MIMIC II considered, the values of \( SC(J) \) were calculated by varying \( J \) from 2 to 5. The final number of clusters corresponded to a local minimum where the difference between the values of the criterion was minor.

In the present work, the maximum limit of variation for the search of the final number of clusters was chosen, with the thought that a smaller number of clusters means a lower number of rules and hence a lower degree of model complexity. For the two databases considered (sonar and readmission datasets) the chosen numbers of cluster were 2.

### 2.2.3 Performance Measures

Traditionally, accuracy has been used to evaluate classifier performance. This measure is defined as the total number of correct classifications over the total number of available samples. Usually, most of the classification problems have two classes, positive and negative cases [38]. Thus, the classified test points can be divided into four categories:

- true positives (TP) - correctly classified positive cases,
- true negatives (TN) - correctly classified negative cases,
- false positives (FP) - incorrectly classified negative cases,
- false negatives (FN) - incorrectly classified positive cases.

Given these categories, the accuracy can be written as (2.4).

\[ accuracy = \frac{TP + TN}{TP + FP + FN + TN} \]  

(2.4)

This criterion is limited, especially in medical applications, for various reasons. If one of the classes is more underrepresented than the others, misclassifications in this class will not have a great impact in the accuracy value. Also, a good classification of a class might be more important than
classifying other classes and this cannot be assessed with accuracy. To take this matter into account, two performance measures were introduced: the sensitivity (2.5) and specificity (2.6):

\[
sensitivity = \frac{TP}{TP + FN} \tag{2.5}
\]
\[
specificity = \frac{TN}{FP + TN} \tag{2.6}
\]

The sensitivity and specificity varies between [0,1].

The receiver operating characteristic (ROC), or simply ROC curve, is a graphical plot which illustrates the performance of a binary classifier system as its discrimination threshold is varied. It is created by plotting the fraction of true positives out of the positives (sensitivity) vs. the fraction of false positives out of the negatives (one minus the specificity), at various threshold settings. An example of ROC curves is shown in Fig. 2.5.

![ROC curve example](image)

Figure 2.5: Example of three ROC curves.

When using normalized units, the area under the curve (AUC) is equal to the probability of a classifier ranking a randomly chosen positive instance being higher than a randomly chosen negative one (assuming 'positive' ranks higher than 'negative'). The AUC measure ranges from 0.5 (random classifier) to 1 (perfect classifier).

In the present work, the Sensitivity and Specificity were used as performance measures for the models using the sonar database and readmission datasets, introduced in section 2.1. However, in the models using sonar database accuracy was used as the main performance measure, and, in the readmission datasets the AUC. This choice was made because of the fact that the two classes of the sonar database had similar numbers of samples (89 samples 45% vs 111 samples 55%) and for the readmission problem the percentage of the class readmitted was only 12.3% against 87.7% of not readmitted. In this case one of the classes is underrepresented and if the accuracy had been used the results would not be realistic, i.e. if a model classified all patients as not readmitted, the accuracy would be ~87.7%, but the AUC measure would be 0.5, corresponding to a random classifier.

For the computation of the measure AUC, only one threshold was used, the one through which the best performance of the model was achieved with the train set. With the resultant sensitivity and
specificity of the test set using that threshold, a point was marked in the ROC. The AUC was computed as the area under the two segments that link the points (0,0) and (1,1) to the point marked with the sensitivity and specificity. By doing this, we ensure a good approximation of the performance of the model.

2.3- Feature Selection

The main characteristic of wrapper methodologies is the involvement of the predictor as part of the selection procedure. In this work, a learning machine was used as a “black box” to score the subsets according to their predictive performance [23]. Wrappers are constituted by three main components:

1) Search method;
2) Learning machine;
3) Feature evaluation criteria.

Wrapper approaches were aimed to improve the results of the specific predictors they work with. During the search, subsets were evaluated without incorporating knowledge about the specific structure of the classification [23].

In section 2.2 the fuzzy modeling technique (learning machine) was introduced. It is considered to have universal function approximation properties, i.e., in theory they could approximate the behaviour of any function. However, as referred in section 1.1.1, in real problems this is rather difficult for a number of reasons, being one of them the high dimensionality of the available data.

Feature selection is generally used to identify which of the available variables are closely related to the prediction of the outcome and to discard those unrelated to it, reducing the dimensionality of the dataset [25, 41, 39]. From the clinical point of view, this process may bring to light new variables that had not been previously considered as relevant to a given outcome.

In the present work, three FS algorithms were applied, the sequential forward selection (SFS), Binary Particle Swarm Optimization (BPSO) and two formulated algorithms: decimal to binary Fish School Search (D2BFSS) and Binary Fish school search algorithms (BFSS).

The following sections, present an overview of the well-known SFS algorithm and the BPSO methods.

2.3.1- Sequential Forward Selection

A detailed description of the sequential forward selection search algorithm used is reported in [39]. Briefly, a model is built for each of the features in consideration, and evaluated using a performance criterion upon the test set. The feature that returns the best value of the performance
criterion is the one selected. Then, other feature candidates are added to the previous best model, one at a time, and evaluated. Again, the combination of features that maximizes the performance criterion is selected. When this second stage finishes, the model has two features. This procedure is repeated until the stop criterion is achieved. In the end, all the relevant features for the considered process should be obtained.

The main advantages of this method relate to its simplicity, possibility of graphical representation of the performance of the added feature and transparent interpretation of the results which, for clinicians, is particular attractive. The main disadvantage is related to the greedy and thus susceptible approach of finding local optima [39].

In this work, unlike the traditional stop criteria of iteration without improving the performance of the models, the maximum number of features selected was used as the stopping criteria. After the maximum number of features had been achieved, the model with the set of features that achieved best performance was considered as the selected best features.

The overall process of the SFS algorithm can be described as:

For each feature in the feature vector X that does not belong to the features of the model:

repeat

Build model using previous features of the model combined with each feature in the feature vector X that does not belong to the features of the model

Compute performance measure;

Select the combination of features with the highest value of AUC as the new features of the model;

until number of selected features reaches defined limit

Select the final features.

The accuracy, for SONAR database, and the AUC, for the MIMIC II derived datasets, were used as performance measures to maximize. For each combination of features selected by the SFS, the process of generating the performance measure of the model can be described by the following steps:

1. The model is trained with the train set and the selected features.
2. With the simulated output of the training set, a threshold is iteratively evaluated in order to find the one who maximise the performance (ACC or AUC, depending on the database).
3. With the threshold found, the test set is then simulated.
4. The final performance of the model is generated with the test set output.
2.3.2- Binary Particle Swarm Optimization

Particle swarm optimization is a stochastic population-based metaheuristic, inspired in swarming behaviour of some biological species (e.g. bird flocks).

There are various ways of encoding a problem solution, the most common and more generic are real, integer and binary encoding. The use of each of them depends on the problem in hand. Normally, in feature selection, the search space organization is made such that each state represents a feature subset [16]. In a problem with \( N_t \) variables, a state is encoded by a sequence of \( N_t \) bits, each bit indicating whether a feature is present or absent. An example of a possible state is represented by the sequence:

\[
x_i = (x_{i1}, x_{i2}, \ldots, x_{iN_t}) = (1, 0, \ldots, 1)
\]  

The variable \( x_j \) corresponds to input \( F_j \), where \( j = 1, \ldots, N_t \). If feature \( F_j \) is to be selected then \( x_{ij} = 1 \) if not \( x_{ij} = 0 \). This process is illustrated in Fig. 2.6.

![Figure 2.6: Decoding process in feature selection, according to [16].](image)

Essentially, in BPSO, each particle is a candidate solution of the optimization problem. A particle is associated to a position and a velocity in the search space, where the method for determining the changes in velocity depends on the particle itself and the other particles.

The iterative process in search of the optimum is [16]:

Step 1: Evaluate each particle in the swarm;
Step 2: Find the swarm and particle best values;
Step 3: Update velocities;
Step 4: Update positions of the particles;
Step 5: Go to Step 1 if not finished/stop criteria.

There are two crucial steps in the way the algorithm operates, the update of velocities and update of particle positions.
1) **Update velocities.** Velocity directs the movement in the search space taking into account the performance of the own particle and of the swarm, and it is update with the following equation:

\[
v_{ij} \leftarrow w v_{ij} + c_1 q (x_{ij}^{pb} - x_{ij}) + c_2 r (x_{ij}^{gb} - x_{ij}) \quad (2.8)
\]

\[i = 1, ..., N, j = 1, ..., N_t\]

The term involving constant \(c_1\) is called the cognitive component and the term involving \(c_2\) is the social component. \(q\) and \(r\) are uniform random numbers \(\in [0,1]\). Once velocities have been update, the restriction \(|v_{ij}| < v_{max}\) is applied; this is a crucial step for the swarm to maintain coherence.

2) **Update particle position.** The logistic function of the velocity is used as the probability distribution for the position, [59]:

\[
\sigma(v_{ij}) = \frac{1}{1+e^{-v_{ij}}} \quad (2.9)
\]

Thus the particle position is calculated for each variable by:

\[
x_{ij} \leftarrow \begin{cases} 
0, & \text{if } r > \sigma(v_{ij}) \\
1, & \text{otherwise}
\end{cases} 
\quad i = 1, ..., N, j = 1, ..., N_t \quad (2.10)
\]

### Objective Function

Recalling that the two main objectives in the FS problem are: maximizing the model accuracy and minimizing the size of the feature subset. The objective function [16] will be defined as a fitness function, being the goal its maximization:

\[
f = \alpha (1 - P) + (1 - \alpha) \left( 1 - \frac{N_f}{N_t} \right) \quad (2.11)
\]

where \(N_f\) is the size of the feature subset and \(N_t\) the total number of features to be selected. The term on the left side of the equation accounts for the overall accuracy or AUC and the term on the right for the percentage of used features. Constant \(\alpha \in [0,1]\) is the weight of the related goal: accuracy or AUC and subset size.
CHAPTER 3

Fish school search

The novel Binary Fish school search algorithm, formulated and presented in this work, was created based on the optimization search algorithm: Fish school search (FSS), invented by C. Bastos Filho and F. Lima Neto in 2007, [17]. In this topic, the original Fish school search optimization algorithm is presented, based on [17], as well as the formulation of the decimal to binary system fish school search algorithm (D2BFSS).

3.1- Original Fish school search

Several oceanic fish species, as well as other animals, present social behaviour. This phenomenon's main purpose is to increase mutual survivability and may be viewed in two ways: (i) for mutual protection and (ii) for synergistic achievement of other collective tasks. Here, protection means reducing the chances of being caught by predators; and synergy, refers to an active mean of achieving collective goals such as finding food.

Apart from debating whether the emergent behaviour of a fish school is due to learning or genetic reasons, it is important to note that some fish species live their entire lives in schools. This reduces individual freedom in terms of swimming movements and increases competition in regions with scarce food. However, fish aggregation is a fact and the benefits largely outweigh the drawbacks.

Along with the development of this technique the authors have taken great care not to depart from the original inspiration source, but FSS contains a few abstractions and simplifications that have been introduced to afford efficiency and usability to the algorithm. The main characteristics derived from real fish schools and incorporated into the core of the approach are sound. They are grouped into two observable categories of behaviours as follows:

- Feeding: inspired by the natural instinct of individuals (fish) to find food in order to grow strong and to be able to breed. Notice that food here is a metaphor for the evaluation of candidate solutions in the search process. An individual fish is considered to be able to lose as well as to obtain weight, depending on the regions it swims in;
• Swimming: the most elaborate observable behaviour utilized in this approach. It aims at mimicking the coordinated and the only apparent collective movement produced by all the fish in the school. Swimming is primarily driven by feeding needs and, in the algorithm, it is a metaphor for the search process itself.

3.1.1-Search Problems and Algorithms

Although there are several approaches for searching, there is, unfortunately, no general optimal search strategy [40]. Thus, solving search problems is sometimes more of an art form than an engineering practice. Although custom-made algorithms are valuable options for specific problems, a more generalized automatic search engine would be a great bonus for tackling problems of high dimensionality. Search problems can be highly varied. For example, they can be classified into two groups with regard to the structure of their search-space: structured or unstructured. For the former, there are many traditional techniques that are, on average, quite efficient. The same observation does not apply to the latter, that is, there is no overall good approach for search spaces on which there is no prior information.

The FSS can be a valuable option for searching in high dimensional and unstructured spaces.

3.1.2-FSS Computational Principles

The search process in FSS is carried out by a population of limited memory individuals – the fishes. Each fish represents a possible solution to the problem. Similar to PSO or GA, search guidance in FSS is driven by the success of some individual members of the population.

The main feature of the FSS paradigm is that all fish contain an innate memory of their successes – their weights. In comparison to PSO, this information is highly relevant because it can obviate the need to keep a log of the best positions visited by all individuals, their velocities and other competitive global variables. Another major feature of FSS is the idea of evolution through a combination of some collective swimming, i.e. “operators” that select among different modes of operation during the search process, on the basis of instantaneous results.

As for dealing with the high dimensionality and lack of structure of the search space, the authors of the algorithm [17], believed that FSS should at least incorporate principles such as the following:

(i) Simple computation in all individuals;
(ii) Various means of storing distributed memory of past computation;
(iii) Local computation (preferably within small radiuses);
(iv) Low communication between neighbouring individuals;
(v) Minimum centralized control (preferably none); and
(vi) Some diversity among individuals.

A brief rationale for the above-mentioned principles is given, respectively: (i) this reduces the overall computation cost of the search; (ii) this allows for adaptive learning; (iii), (iv) and (v) these keep computation costs low as well as allowing some local knowledge to be shared, thereby speeding up convergence; and finally, (vi) this might also speed up the search due to the differentiation/specialization of individuals. These principles, incorporated in FSS, led the authors to believe that FSS could deal with multimodal problems.

3.1.3-Overview of the algorithm

The inspiration mentioned, together with the principles just stated above, were incorporated in the approach in the form of two operators that comprise the main routines of the FSS algorithm. To understand the operators, a number of concepts must be defined.

The concept of food was considered as related to the function to be optimized in the process. For example, in a minimization problem the amount of food in a region would be inversely proportional to the function evaluation in this region. The “aquarium” is defined by the delimited region in the search space where the fish can be positioned. The operators were grouped in the same manner in which they were observed when drawn from the fish school, defined as follows:

- Feeding: food is a metaphor for indicating to the fish the regions of the aquarium that are likely to be good spots for the search process;
- Swimming: a collection of operators that are responsible for guiding the search effort globally towards subspaces of the aquarium that were collectively sensed by all individual fish as more promising with regard to the search process.

3.1.4-The Feeding Operator

As in real situations, the fish of FSS are attracted to food scattered in the aquarium in various concentrations. In order to find greater amounts of food, the fish in the school can move independently (see individual movements in the next section).

As a result, each fish is allowed to grow in weight, depending on its success or failure in obtaining food. The authors proposed that fish’s weight variation be proportional to the normalized difference between the evaluation of fitness function of previous and current fish position with regard to food concentration of these spots. The assessment of ‘food’ concentration considers all problem dimensions, as shown in (3.1):
\[ W(t + 1) = W_i(t) + \frac{f[x_{i}(t+1)] - f[x_{i}(t)]}{\max \{f[x_{i}(t+1)] - f[x_{i}(t)]\}} \]  

(3.1)  

where \( W_i(t) \) was the weight of the fish \( i \), \( x_{i}(t) \) the position of the fish \( i \) and \( f[x_{i}(t)] \) evaluated the fitness function (i.e. amount of food) in \( x_{i}(t) \).

A few additional measures were included to ensure rapid convergence toward rich areas of the aquarium, namely:

- Fish weight variation is evaluated once at every FSS cycle;
- An additional parameter, named weight scale (\( W_{\text{scale}} \)) was created to limit the weight of a fish. The fish weight may vary between 1 and \( W_{\text{scale}} \).
- All the fish are born with weight equal to \( \frac{W_{\text{scale}}}{2} \).

### 3.1.5-The Swimming Operators

A basic animal instinct is to react to environmental stimulation (or sometimes, the lack of it). In this approach, swimming is considered to be an elaborate form of reaction regarding survivability. In FSS, the swimming patterns of the fish school are the result of a combination of three different causes (i.e. movements).

For fish, swimming is directly related to all the important individual and collective behaviours such as feeding, breeding, escaping from predators, moving to more liveable regions of the aquarium or, simply being gregarious. This panoply of motivations to swim away inspired the authors [17] to group causes of swimming into three classes: (i) individual, (ii) collective-instinct and (iii) collective volition. Below further explanations on how computations are performed on each of them are provided.

### 3.1.6-Individual Movement

Individual movement occurs for each fish in the aquarium at every cycle of the FSS algorithm. The swim direction is randomly chosen. Provided the candidate destination point lies within the aquarium boundaries, the fish assess whether the food density there seems to be better than at its current location. If not, or if the step-size would be considered not possible (i.e. lying outside the aquarium or blocked by, say, reefs), the individual movement of the fish would not occur. Soon after each individual movement, feeding would occur, as detailed above.

For this movement, a parameter was defined to determine the fish displacement in the aquarium called individual step (\( \text{step}_{\text{ind}} \)). Each fish moves \( \text{step}_{\text{ind}} \) if the new position has more food than the previous position. Actually, to include more randomness in the search process the individual step is multiplied by a random number generated by a uniform distribution in the interval \([-1, 1]\).
represented as $u$ in (3.1). In this simulation, the individual step was decreased linearly in order to provide exploitation abilities in later iterations:

$$
\ddot{x}(t + 1) = \ddot{x}(t) + u(-1,1)S_{\text{ind}}(t),
$$

$$
S_{\text{ind}}(t) = \text{step}_{\text{ind, initial}} - (\text{step}_{\text{ind, initial}} - \text{step}_{\text{ind, final}}) \frac{g_{\text{actual}}}{g_{\text{final}}}
$$

where $g_{\text{actual}}$ is the number of the actual iteration and $g_{\text{final}}$ is the total number of iterations.

Fig. 3.1 shows an illustrative example of this swimming operator. One can note that just the fish that found spots with more food have moved.

3.1.7-Collective-Instinctive Movement

After the individual movement, a weighted average of individual movement based on the instantaneous success of all fish of the school is computed. This means that fish that had successful individual movements influence the resulting direction of movement more than the unsuccessful ones. When the overall direction is computed, each fish is repositioned. This movement is based on the fitness evaluation enhancement achieved, as shown in (3.3).

$$
x_i(t + 1) = x_i(t) + I(t), \quad I(t) = \frac{\sum_{i=1}^{N} \Delta x_{\text{ind}}[f[x_i(t+1)] - f[x_i(t)]]}{\sum_{i=1}^{N} [f[x_i(t+1)] - f[x_i(t)]]}
$$

where $\Delta x_{\text{ind}}$ is the displacement of the fish $i$ due to the individual movement in the FSS cycle. Fig. 3.2 shows the influence of the collective-instinctive movement in the example presented in Fig. 3.1. One can note that in this case all the fish had their positions adjusted.
After individual and collective-instantaneous movements are performed, one additional positional adjustment is still necessary for all fish in the school: the collective-volatile movement. This movement is devised as an overall success/failure evaluation based on the incremental weight variation of the whole fish school. In other words, this last movement will be based on the overall performance of the fish school in the iteration.

The rationale is as follows: if the fish school is putting on weight (meaning the search has been successful), the radius of the school should contract; if not, it should dilate. This operator is deemed to help greatly in enhancing the exploration abilities in FSS. This phenomenon might also occur in real swarms, but the reasons are as yet unknown.

The fish-school dilation or contraction is applied as a small step drift to every fish position with regard to the school’s barycenter. The fish-school’s barycenter is obtained by considering all fish positions and their weights, as shown in (3.4).

Collective-volatile movement will be inwards or outwards (in relation to the fishschool’s barycenter), according to whether the previously recorded overall weight of the school has increased or decreased in relation to the new overall weight observed at the end of the current FSS cycle.

$$Bari(t) = \frac{\sum_{i=1}^{N} x_i(t) w_i(t)}{\sum_{i=1}^{N} w_i(t)}$$  \hspace{1cm} (3.4)

For this movement, a parameter called volitive step ($step_{vol}$) was defined as well. The new position is evaluated as in (3.5) if the overall weight of the school increases in the FSS cycle; if the overall weight decreases, (3.6) should be used.

$$x_i(t + 1) = x_i(t) - step_{vol} \cdot rand \cdot [x_i(t) - Bari(t)]$$  \hspace{1cm} (3.5)
\[ x_i(t + 1) = x_i(t) + \text{step}_{vol}. \text{rand} \cdot [x_i(t) - \text{Bari}(t)] \]  

where \( \text{rand} \) is a random number uniformly generated in the interval \([0,1]\). We also decreased the linear \( \text{step}_{vol} \) along the iterations.

Fig. 3.3 shows the influence of the collective-volitive movement in the example presented in Fig. 3.1 after individual and collective-instinctive movements. In this case, as the overall weight of the school had increased, the radius of the school diminished.

![Figure 3.3: Collective-volitive movement is illustrated here before and after its occurrence; pink dots are fish positions after and green dots are the same fish before collective-volitive movement. The position of the barycentre is represented by the blue dot.](image)

3.1.9-FSS Cycle and Stop Conditions

The FSS algorithm starts by randomly generating a fish school according to parameters that control fish sizes and their initial positions.

Regarding dynamic, the central idea of FSS is that all bio-inspired operators perform independently from each other. The FSS search process is enclosed in a loop, where invocations of the previously presented operators will occur until at least one stop condition is met. Stop conditions conceived for FSS are as follows: limitation of the number of cycles, time limit, maximum school radius and maximum school weight.

Below, the pseudo-code for the Fish School Search Algorithm is presented. In the initialization step, each fish in the swarm has its weight initialized with the value \( \frac{w_{\text{scale}}}{2} \) and its position in each dimension initialized randomly in the search space.
**Algorithm Fish School Search**

1. Initialize fish in the swarm
2. **While** maximum iterations or stop criteria is not attained **do**
3. **for** each fish i in the swarm **do**
   a. **update position applying the individual operator**
      \[ \Delta x_i(t + 1) = step_{ind}(t) \cdot 2 \cdot rand \cdot direction \]
      \[ temp_i = x_i(t) + \Delta x_i(t + 1) \]
      calculate fish fitness \( f_i(temp_i) \)
      **if** \( f(temp_i) < f(x_i(t)) \)
      \[ x_i(t + 1) = temp_i \]
      \[ f_i(t+1) = f_i(temp_i) \]
      **else**
      \[ x_i(t + 1) = x_i(t) \]
      \[ f_i(t+1) = f_i(t) \]
   b. **apply feeding operator**
      update fish weight according to (3.1)
   c. **apply collective-instinctive movement**
      update fish position according to (3.3)
   d. **apply collective-volitive movement**
      **if** overall weight of the school decreases in the cycle
      update fish position using (3.5)
      **elseif** overall weight of the school decreases in the cycle
      update fish position using (3.6)
4. **end for**
5. **decrease the individual and volitive steps linearly**
6. **end while**

### 3.1.10-Illustrative Example

This section presents an illustrative example (presented in [17]) aimed at better understanding of how FSS can be used and, ultimately, how it works. The selected example considers a small school and a very simple problem that is three fish are set to find the global optimum of the sphere function in two dimensions. The sphere function is presented in (3.7) and its parameters are: (i) feasible space [-10,10], (ii) number of iterations equal to 10, (iii) \( W_{scale} = 10 \), (iv) initial \( step_{ind} = 1 \), (v) final \( step_{ind} = 0.1 \), (vi) initial \( step_{vol} = 0.5 \), (vii) final \( step_{vol} = 0.05 \). Table 3.1 includes initial values associated with the experimental fish school; Fig. 3.4a presents start-up loci of all fish.

\[
F_{sphere}(x) = \sum_{i=1}^{n} (x_i)^2 \quad (3.7)
\]
After initialization, all fish are free to check for new candidate positions that are generated by the individual movement operator. Assuming that these positions were $x_1 = (9.6,6.2)$, $x_2 = (4.6,4.4)$ and $x_3 = (6.2,4.2)$, and the associated fitnesses $f(x_1) = 130.6$, $f(x_2) = 40.52$ and $f(x_3) = 56.08$, one should notice that fish #2 and fish #3 found best positions, whereas fish #1 did not move. The positions after the individual movement were then $x_1 = (9.7)$, $x_2 = (4.6,4.4)$ and $x_3 = (6.2,4.2)$. Fig. 3.4b illustrates the individual movement of the three fish in search space for the sphere problem.

According to this model, the next operator to be computed should be feeding. As fish #1 remained in the same position, it would not change its weight. The weight of fish #2 and fish #3 would change according to (3.1). The weight variation depends on the maximum fitness change. The maximum fitness variation in this case was achieved by fish #3 and is equal to 23.92. As a result, fish #3 increased its weight by 1 unit and its new weight became 6. The fitness variation of fish #2 was 20.48. Dividing the fitness variation of fish #2 by maximum fitness change, concluding that the weight variation of fish #2 is 0.86. The new weight of fish #2 is then 5.86. Following the model, the third operator to be computed would be the collective instinctive one. This operator evaluates the collective displacement of the fish school considering the individual fitness variations and the individual movement according to (3.3). As fish #1 stayed in the same position, it would not influence the overall calculation. Considering the values obtained in this iteration, the displacement was (-1.2,-0.6). This vector applies to all the fish (including fish #1), so the new positions, after third operator computations, were $x_1 = (8.4,5.6)$, $x_2 = (3.4,3.8)$ and $x_3 = (5,3.6)$.

Then the fitnesses, regarding new positions recalculation, were 101.8, 26 and 37.96 for fish #1, #2 and #3, respectively. The individual displacement of all fish due to collective-instinctive operator is presented in Fig. 3.4c. The interested reader may find it interesting to compare Fig. 3.4b and Fig. 3.4c.

The last operator to be considered in this example is the collective-volitive one. For that, one has to obtain the instantaneous value of the barycenter of the fish school according to (3.4). In this case, the barycenter was (4.96,4.25). Notice that the weight of whole school has increased, therefore a contraction instead of a dilatation was the implicit decision of the school (i.e. collective-volitive). By means of using (3.5), the new positions were $x_1 = (5.81,4.89)$, $x_2 = (4.02,3.98)$ and $x_3 = (4.98,3.92)$. The barycentre and the collective-volitive movement for this step are presented in Fig. 3.4d.

At this point, the algorithm tests if valid stop-conditions are met. Obviously it was not the case yet, thus a new cycle began as explained above. If one compares the initial and final positions

<table>
<thead>
<tr>
<th>Initial conditions</th>
<th>Fish</th>
<th>weight</th>
<th>position</th>
<th>fitness</th>
</tr>
</thead>
<tbody>
<tr>
<td># 1</td>
<td>5</td>
<td>(9,7)</td>
<td>130</td>
<td></td>
</tr>
<tr>
<td># 2</td>
<td>5</td>
<td>(5,6)</td>
<td>61</td>
<td></td>
</tr>
<tr>
<td># 3</td>
<td>5</td>
<td>(8,4)</td>
<td>80</td>
<td></td>
</tr>
</tbody>
</table>
illustrated in Fig. 3.4, after this first iteration, the reader can observe that all fish are closer to the optimum point (0,0).

Of course the optimum point is unknown to the algorithm. However, in a very peculiar manner the FSS model assures fast convergence towards it (i.e. the goal for the search process) because of the above mentioned natural principles instantiated in the FSS algorithm.

Figure 3.4: Example with three fish in the sphere example: (a) Initial position, (b) individual movement, (c) instinctive collective movement and (d) collective-volitive movement.
In order to illustrate the convergence behaviour of the fish school along the iterations, the simulation results for the sphere function are presented. For these simulations we used 30 fish, \([-100,100]\) in the two dimensions, initialization range \([0,100]\) in the two dimensions, \(w_{\text{scale}} = 500\), initial step\(_{\text{ind}} = 1\), final step\(_{\text{ind}} = 0.1\), initial step\(_{\text{vol}} = 0.5\), final step\(_{\text{vol}} = 0.05\). Fig. 3.5 shows the fish positions after iteration (a) 1, (b) 5, (c) 10, (d) 20, (e) 30, (f) 40, (g) 50, (h) 100 e (i) 200, respectively. One can note that the school was attracted to the optimum point (0,0).
3.2- Decimal to binary Fish school search

The FSS algorithm, described in section 3.1, is a high dimension search optimization process in continuous space. The first intuitive and logical approach was to use the FSS continuous algorithm to search a one dimension integer number that would then be transformed, in the objective function, in a vector of binary input with the dimension equal of the number of the features to be selected. To do so, the decimal to binary system representation was chosen.

This vector of binary input would be then be used as the encoding of the BPSO algorithm, presented in section 2.3.2.

This relatively simple representation allows the original algorithm, to select features without major changes in the original algorithm. All the operators were used as described in section 3.1. The only modification needed was to round the position of each fish to its nearest integer. Thus, it was only necessary to use one dimension on the search space to search the decimal system representation of the solution. This approach was called the D2BFSS algorithm.

3.2.1-Objective function

In order to evaluate the fitness of the decimal system solution (position of the fish), the integer solution was transformed to its binary representation, which was a vector of 0 or 1 bits with dimension equal to the maximum number of features to be selected.

Inspired by the objective function used on [16], presented in section 2.3.2, the following fitness function was used to describe the performance of the selected features during the FS process:

\[ f = \alpha(1 - P) + (1 - \alpha) \left( 1 - \frac{N_f}{N_t} \right) \]  \hspace{1cm} (3.7)

where \( N_f \) represents the number of features selected and \( N_t \) the total number of features while the value \( P \) accounts for the performance measure of the test set. The \( \alpha \) value varies between 1 and 0. If the right side of (3.7) was not used (\( \alpha = 1 \)), there would not be a restriction to the number of features selected by the algorithm.
CHAPTER 4

Binary Fish School Search

It is important to note that the D2BFSS approach, section 3.2, does not manipulate the vector of bits (feature selected) in its internal mechanisms (the FSS movement operators). Thereby, the decimal to binary system approach may have problems with convergence, low performance or even be a random search.

With this concern in mind, it was decided to modify the internal mechanisms of the FSS algorithm to manipulate binary inputs himself. The following sections describe the modifications to the fish school search algorithm, emerging the binary fish school search.

4.1- Encoding

There are various ways of encoding a problem solution, the encoding presented here was inspired in [16], similar to section 2.3.2. An example of a possible state (position of a fish) is represented by the sequence:

\[ x_i = (x_{i1}, x_{i2}, \ldots, x_{iN_i}) = (1, 0, \ldots, 1) \quad (4.1) \]

Where \(N_i\) is the total number of features to be selected. Each bit indicates whether or not a feature is selected. This binary scheme, offers a straightforward representation of a feature subset, allowing the algorithm to search through the workspace, adding or removing features, simple by flipping bits in the sequence.

While the FSS algorithm was not originally developed in the context of binary encoding, it appeared to be possible to modify the real to a binary encoding, keeping the following principles:

- to follow the internal mechanisms of the original algorithm, without losing the meaning of each operator;
- to add few additional parameters;
- to ensure the convergence of the algorithm;
- to keep simplicity and understanding to the modifications.
In the next sections, the modifications made to each of FSS internal mechanisms are presented.

### 4.2-Initialization

For each fish $i$, the initial position was initialized randomly by doing:

$$x_{ij} \left\{ \begin{array}{ll} 1, & \text{if } r > 0.5 \\ 0, & \text{otherwise} \end{array} \right. \quad i = 1, ..., N, \ j = 1, ..., N_t$$  \hspace{1cm} (4.2)


where $r$ is a random number uniformly generated in the interval $[0,1]$, $N$ the number of fishes and $N_t$ the total number of features to be selected.

By doing this, the algorithm starts with completely random positions, being the number of features selected at the start around $Nt/2$. If the initial number was too small, the algorithm might not converge freely along iterations.

### 4.3-Individual Movement

The Individual movement occurs once in every cycle of the BFSS. For each fish $i$, and for each bit $j$, if a random number $k$ (uniform distribution in the interval $[0,1]$), is smaller than $S_{\text{ind}}(t)$ the bit will flip, otherwise it will not change:

$$x_{ij} \left\{ \begin{array}{ll} \bar{x}_{ij}, & \text{if } k < S_{\text{ind}}(t) \\ x_{ij}, & \text{otherwise} \end{array} \right. \quad i = 1, ..., N, \ j = 1, ..., N_t$$  \hspace{1cm} (4.3)

Parameter $S_{\text{ind}}$, in the same way as the FSS, will decrease linearly along the iterations depending on the first value and the last value of step$_{\text{ind}}$. This allows a soft convergence through the iterations.

A fish will move if the new position has more food than the previous position, i.e. if the fitness function of new set of features selected (new position) has a better performance than the previous one. By doing this, the random exploration of each individual fish is preserved.

### 4.4-Collective-Instinctive Movement

After the individual movement, the weighted average of the individual movements, based on fishes that had moved, is calculated. This process was executed in the same way as the FSS, equation (3.1).

In order to make all fishes head to the direction of the successful individual movement position some changes had to be made to the original FSS algorithm.
When dealing with positions with bits (values 0 or 1), equation (3.3) loses its meaning. The displacement of the fish, $\Delta x_{\text{ind}}$ in equation (3.3), can no longer be quantified correctly using the discrete flipping of a bit.

For that reason, equation (4.4) was used to describe the resultant position of the overall successful of the individual movement:

$$\bar{I}(t) = \frac{\sum_{i=1}^{N} x_{\text{ind}} \Delta f_i}{\sum_{i=1}^{N} \Delta f_i} \quad (4.4)$$

In (4.4), $\Delta x_{\text{ind}}$ in (3.3) was replaced with $x_{\text{ind}}$. In this approach, the use of the actual position of the fishes that had success in the individual movement is seen as being more descriptive than the flipping of bits.

The resulting vector $\bar{I}$ has the same dimension as the positions of the fishes, but with values varying between 0 and 1. As an illustrative example, (4.5) represents a possible configuration of $\bar{I}$:

$$\bar{I} = [0.1 \ 0.5 \ 0 \ 0.3 \ \ldots \ 0.7] \quad (4.5)$$

The goal of the Collective-Instinctive Movement operator is to attract each fish to the resultant direction of the individual movement operator. In the Binary Fish School Search each fish approaches $\bar{I}$. To do so, it is necessary for it to have bit format, two options were here considered to transform $\bar{I}$ in a bit vector:

a) Using a constant threshold in all iterations– if the values of the bits of $\bar{I}(t)$ were below the parameter $\text{thres}_{\text{c}}$, they would be considered 0, otherwise 1.

For example, if the value of $\text{thres}_{\text{c}} = 0.4$ was used in the example (4.5), the resultant vector would be:

$$\bar{I} = [0 \ 1 \ 0 \ 0 \ \ldots \ 1] \quad (4.6)$$

The problem of using a constant threshold in all iterations is that, depending on the evolution of the FS process, $\bar{I}(t)$ could be formed of only 0s, i.e. all the values of $\bar{I}(t)$ lower than $\text{thres}_{\text{c}}(t)$. In addition, if in any iteration the algorithm favoured a certain feature, it could happen that the algorithm loses the exploration abilities in later iterations. If this occurred, it would introduce trends and convergence to local maxima.

b) Using an adaptive threshold for each iteration: multiplying the parameter $\text{thres}_{\text{c}}$ by the max value of $\bar{I}(t)$. The resultant value of this multiplication would then be used as threshold in the current iteration for this operator.

For the example (4.5), if the parameter $\text{thre}_{\text{c}}$ was 0.4, the threshold used in this iteration would be $0.4 \times 0.7 = 0.28$, considering 0.7 the max value of $\bar{I}$ (4.5), resulting in:

$$\bar{I} = [0 \ 1 \ 0 \ 1 \ \ldots \ 1] \quad (4.7)$$
Therefore in b), for each iteration, the threshold to compute $I(t)$ binary vector was calculated using the max value of $\tilde{I}(t)$. This allowed the algorithm to select at least 1 feature, less likely to incurring the problem described in the option a).

The study to these two options is presented in chapter 5.

After the computation of $\tilde{I}(t)$ in bit format, all fish position could now tend to $\tilde{I}(t)$. To do so, the position of each fish was compared with $\tilde{I}(t)$. One bit (randomly chosen) of the fish that did not have the same value as $\tilde{I}(t)$ was flipped. This process approaches the position of each fish to $\tilde{I}(t)$. In comparison with the original algorithm, $\tilde{I}(t)$ no longer represents the direction but the position resultant of the successfully individual movements.

By only flipping one bit per fish, a soft and steady convergence of the algorithm is expected. An illustrative example can be represented:

$$x(t) = [0 \ 1 \ 0 \ 1 \ 1] \rightarrow I = [0 \ 1 \ 0 \ 0 \ 0]$$

$$x(t + 1) = [0 \ 1 \ 0 \ 0 \ 1]$$

In (4.8) the fish $x(t)$ moved in the direction of $\tilde{I}(t)$. The bits with the same values of $\tilde{I}(t)$ are represented in red. The resultant position, $x(t + 1)$, is achieved by flipping one random bit that has different values, represented in green. The total number of bits in red in the new position is greater than the one in the position before the collective-instinctive movement, making the new position of the fish to be closer to $\tilde{I}(t)$.

### 4.5-Collective-volitive Movement

Similarly to the Collective-Instinctive Movement operator, the Collective-volitive operator underwent some changes. The main goal of this operator is, depending of the success of the individual movement, to contract or dilate the fish position to or from the barycentre.

The barycentre was computed in the same way as in the FSS algorithm (3.4). Analogously to the computation of the vector $\tilde{I}(t)$, after (3.4) the barycentre was not obtained in a bit format. Thereby, two options were also considered to transform the barycentre to a bit format:

a) Using a constant threshold through iterations: $thresh_v$

b) Using the adaptive threshold for each iteration: multiplying $thresh_v$ with the max value of barycentre.

If the overall individual movement was a success (overall weights improved in the iteration) each fish would approximate to the barycentre. Similarly to the process in the Collective-Instinctive Movement operator, section 4.4, every bit per fish were compared to the barycentre. One bit (chosen randomly) that was not the same value as the barycentre was then flipped. By making only one flip per fish, the algorithm enables a soft directing from the previous position to the new one, closer to the
barycentre. An illustrative example to the case of the improvement of the overall weights (contraction) is shown:

\[
x(t) = [0 \ 1 \ 0 \ 1 \ 1] \rightarrow bari = [0 \ 1 \ 0 \ 0 \ 0] \\
x(t + 1) = [0 \ 1 \ 0 \ 0 \ 1]
\]  \quad (4.9)

In (4.9), fish \(x(t)\) changed randomly one of its bits that were different (green) from barycentre (\(bari\)). This allowed the fish to approximate to the baricenter.

If the overall weights had not improved, each fish has to move to the opposite direction of the barycentre. To do this, the concept of anti-barycenter is introduced, consisting of a vector with the same dimensions as the barycentre but with flipped bits. In this situation, the process is the same as described above for the case of contraction to the barycentre but using the anti-barycentre. In (4.10) the representation of the case of not improvement of the overall weights of the example (4.9) is presented:

\[
x(t) = [0 \ 1 \ 0 \ 1 \ 1] \rightarrow antibari = [1 \ 0 \ 1 \ 1 \ 1] \\
x(t + 1) = [1 \ 1 \ 0 \ 1 \ 1]
\]  \quad (4.10)

In (4.10), the fish new position \(x(t + 1)\) is obtained comparing each bit with the anti-baricenter of the barycentre presented in (4.9). One of the bits with different values (green), was flipped, making the new position of the fish to be closer to the \(antibari\) and consequently further to \(bari\) in (4.9).

With the one bit flip mechanism, the barycentre could no longer be seen as a possible solution (as is FSS algorithm) but as a point of reference to guide the fishes in the contraction or dilation process. The best solution per iteration would now be selected by the fish with the best performance after the collective-volatile movement.

After the collective-volatile movement, a new cycle begins.

4.5-Objective function

Although some of the parameters of the BFSS algorithm influence the final number of features selected (use of thresholds), the process of developing an objective function is critical, since it serves as guidance in search of the optimum.

The fitness function was defined as in [16], being the goal its maximization. The most suitable representation to the proposed task is shown:

\[
f = \alpha (1 - P) + (1 - \alpha) \left(1 - \frac{N_f}{N_t}\right)
\]  \quad (4.11)

where \(P\) is the classifier performance measure (ACC or AUC, depending on the database), \(N_f\) the number of features selected and \(N_t\) is the total number of features to be selected. The term on the left side of the equation accounts for the overall accuracy of the model while the term on the right for the percentage of used features. Note that both terms in the objective function are normalized. Constant
$\alpha \in [0,1]$ defines the weight of the related goal, performance and subset size. The constant $\alpha$ is a parameter of the algorithm and varies depending the total number features to be selected ($N_t$) and the desired number of features selected.

### 4.7-Parameters

The choice of the set of parameters is a crucial step in wrapper search methods. If the set is not the most suitable, the predictor will underperform, which might mislead the search algorithm.

When performing the modifications presented above, some parameters were introduced. Fig. 4.1 summarises the set of parameters used in the FSS, D2BFSS and BFSS algorithms.

<table>
<thead>
<tr>
<th>FSS</th>
<th>D2BFSS</th>
<th>BFSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>• No. of fishes</td>
<td>• No. of fishes</td>
<td>• No. of fishes</td>
</tr>
<tr>
<td>• No. of iterations</td>
<td>• No. of iterations</td>
<td>• No. of iterations</td>
</tr>
<tr>
<td>• $W_{\text{scale}}$</td>
<td>• $W_{\text{scale}}$</td>
<td>• $W_{\text{scale}}$</td>
</tr>
<tr>
<td>• $\text{step}_{\text{ind}}$ [initial and final]</td>
<td>• $\text{step}_{\text{ind}}$ [initial and final]</td>
<td>• $\text{step}_{\text{ind}}$ [initial and final]</td>
</tr>
<tr>
<td>• $\text{step}_{\text{vol}}$ [initial and final]</td>
<td>• $\text{step}_{\text{vol}}$ [initial and final]</td>
<td>• $\text{thres}_c$</td>
</tr>
<tr>
<td>• $\text{thres}_v$</td>
<td>• $\text{thres}_v$</td>
<td>• $\alpha$</td>
</tr>
</tbody>
</table>

Figure 4.1: Parameters for the original FSS, the decimal to binary FSS and the binary FSS

It is known that the more parameters an algorithm uses, the more time is taken for parameters estimation and the greater the complexity in the process. The approach taken, as well as the resultant set of parameters selected, is expected to be able to achieve convergence, and although in a more subjective way, maintaining the meaning of each operator in the original FSS algorithm.

#### 4.8- BFSS cycle and stop condition.

In the same way as the FSS algorithm, the BFFS starts by randomly generating a fish school (features selected). In general, the cycle is similar to the FSS, being the main differences the modifications to each internal mechanism (operators). In addition, instead of using the position of the barycentre as the best solution in the iteration, the BFFS uses the fish with the maximum fitness function.

Regarding the stopping criterion, the followings could be used: time limit, maximum school weight and maximum number of iteration reached (used in all the experiments here presented).
Chapter 5

Results

The main objective of this chapter is to evaluate the applicability of the proposed search optimization algorithms. These methods combine the machine learning algorithm, introduced in section 2.2, with the state of the art search algorithms presented in section 2.3 and the formulated in chapters 3 and 4, using the approach described in the section 5.1. They will be compared with each other and with the results obtained without FS, based on the predictive performance and the number of selected features.

For each of the two databases considered in this work, a study was made, so as to ascertain the parameters of the optimization algorithms to the feature selection problem.

The implementation of the algorithms and the obtainment of the results were made with Matlab ® R2010a.

5.1-Description of the approach

The use of a learning machine in wrapper methods, so as to evaluate subset suitability, involves a correct feature subset assessment. The process described in this section, was preformed for each of the databases considered in this work.

The data was firstly divided in two groups, the feature selection (FS) subset and the model assessment (MA) subset. This division was random but with the same percentage of each class in each subset, i.e. 50% of the samples belong to the FS and the other 50% to the MA subset, and both groups had the same percentage of samples for each class considered.

The FS subset was divided in 70% of the samples for training set and 30% for testing set. This division was also performed randomly and with the same percentage of each class in each set. The feature selection was then accomplished and, after the stop criterion was reached, the model with the best performance was selected. The features selected, as well as its threshold, were then recorded and a 10-fold cross validation was performed to the MA subset.
The k-fold cross validation consists of dividing data into k subsets, using at each time k−1 for training and the other set for testing. Each subset had been divided to have the same percentage of samples for each class. For k times, models were trained and tested using the recorded features and threshold, until all possible combinations of training and testing sets were covered. The results are the average of the performance measure, introduced in section 2.2.3, on all splits. The mean and the standard deviation of the AUC, sensitivity and specificity and accuracy are then reported. The k-fold cross validation allows the evaluation of the validity and robustness of the discovered model by assessing how the resulting model from feature selection would generalize to an independent data set (MA subset).

To reduce variability, introduced by the division of the data not only in the FS/MA subset but also in the division of train/test subsets, 10 rounds of the 10-fold cross validation process were performed always using different partitions. The round with the best performance was selected, and the performance measures of that round described the model created with the selected set of features. Fig. 5.1 summarizes the process.

5.2-Optimization Parameters

In order to choose an appropriated set of parameters to the optimization algorithm, it was important to do a study of the parameters to be used. Recalling that D2BFSS and BFSS were never tested before, several measures were chosen to select a fair set of parameters and to evaluate the internal dynamic of the proposed algorithms. These measures, that will be called indicators, used the results of the best solution in the FS process and also the MA results:

- **FS best fitness**: encompassing the performance of the best model and the number of features selected in all iterations of the FS process. It ranged from 0 to 1, the higher the better. Calculated with (2.11), (3.7) or (4.11) depending on the FS algorithm.
- **Number of feature selected**: the lower the better.
• **Performance of the best model in the FS**: the performance of the best model in the FS process, the ACC in the case of the sonar base and the AUC in the readmission databases. The higher the better.

• **Performance of the MA**: the 10-fold cross validation result using the MA subset with the features selected using the FS algorithm. The mean ACC for the sonar database and mean AUC for the readmission databases. The higher the better.

• **Iteration of the optimization algorithm with the best solution**: Although it can occur, the optimization algorithm is not supposed to find the best solution in the first’s iterations. The algorithm should evolve and converge to a better solution. Can vary between 1 and the max iterations used in the FS process. Low values for this indicator are considered lower quality solutions.

• **Percentage of contraction in all iterations**: the percentage of all iterations in which, in the collective-volitive operator, the algorithm contract. This allows the analysis of the internal behaviour of the algorithm. If the percentage is 0%, the algorithm only expanded and with 100% the algorithm contract in all iterations. Neither 0% nor 100% are favourable to the correct execution of the algorithm, to the general level of convergence (0%) and convergence to local maxima (100%).

• **Number of repetitions of the same position of the barycentre**: this measure allows the verification of correct function of the internal mechanisms of the algorithm. Varies between 0 and the maximum number of iterations. The limits are considered as lower quality solutions.

• **The plot**: the result of the visual analysis of the graphical evolution of the best solution per iteration in the FS process. This graph is supposed to show the convergence of the algorithm along its iterations, as well the oscillation near the local maxima. Classified as − (bad conjugation), + (good) and ++ (very good).

The optimization algorithm should search the space for the solution (exploration) and, in the same time, converge to a good solution (exploitation). The three last measures help the algorithm developer to understand what is happening in the internal dynamics of the algorithm, and to achieve a good ration of exploration and exploitation.

The D2BFSS and BFSS algorithms, here formulated, do not guarantee in advance a convergence evolution or a good performance of the model created so, it is crucial to consider as many factors as possible to ensure the correct function of the algorithm.
5.3-Sonar database

As described in section 2.1, the sonar database consists of 208 samples with 60 features and two classes. As the number of samples for each class was nearly the same, the main performance measure to be maximised was the accuracy. Due to the number of features to be selected and the number of samples it was decided that, after the study of the optimization parameters, 100 rounds of the whole process (FS+MA) would be simulated with different partitions of FS/MA. The mean, the standard deviation and the best model created, in the 100 rounds, would then be used to compare the different feature selection algorithms. The same partitions of the data were used for the 100 rounds between the different algorithms, so that the comparison of the results would be fair.

5.3.1-Sequential Forward Selection

After consulting [16,21], it was verified that previous studies that used the SFS applied to the sonar database did not select more than 15 features. The SFS stop criteria was chosen as 15 features selected.

Since the SFS does not have parameters, a study of the parameters was not necessary, and 100 rounds of the process described in section 5.1 were computed. Table 5.1 shows the results of the mean and standard deviation results of the 10-fold cross validation of the 100 rounds and also the model with better performance. The selected threshold and number of features in each round were also recorded.

Table 5.1: Model assessment results - SFS method using sonar database.

<table>
<thead>
<tr>
<th>100 rounds</th>
<th>mean</th>
<th>sensitivity</th>
<th>specificity</th>
<th>mean</th>
<th>sensitivity</th>
<th>specificity</th>
<th>threshold</th>
<th>features selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC</td>
<td>ACC</td>
<td></td>
<td></td>
<td>AUC</td>
<td>ACC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.73</td>
<td>73.19</td>
<td>0.69</td>
<td>0.12</td>
<td>12.05</td>
<td>0.18</td>
<td>0.21</td>
<td>0.48</td>
<td>8</td>
</tr>
<tr>
<td>std</td>
<td>0.04</td>
<td>0.07</td>
<td>0.03</td>
<td>2.59</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>4</td>
</tr>
<tr>
<td>best round</td>
<td>0.81</td>
<td>81.51</td>
<td>0.75</td>
<td>15.13</td>
<td>0.13</td>
<td>0.24</td>
<td>0.45</td>
<td>9</td>
</tr>
</tbody>
</table>

The SFS algorithm allows the visualization of the quality of each feature subset selected by the algorithm, Fig. 5.2 summarizes graphically the process of FS for the best model of the 100 rounds. After the addition of the ninth feature (marked in green in Fig. 5.2), the performance of the models created with the additional feature didn't improve.
As described in section 4.3.2, the D2BFSS algorithm had the same parameters as the Fish School Search genetic algorithm, with the addition of the parameters $\alpha$ (see Fig. 4.1). The selected stop criterion was the number of iterations: 300 iterations.

To maintain consistency in the comparison of the results between different FS algorithms, the same number of fish/particles (30) and the same partition of the data (FS/MA and train/test) was used, as well as the same stop condition for the search process.

In every study of the parameters the same initial position of the fishes/particles was utilized. This reduced the variability of the results and ensured a coherent analysis of the parameters of the optimization algorithm.

According to the examples in [17], the most sensitive parameters of the FSS algorithm are the step$_{\text{ind}}$ (initial and final) and the step$_{\text{vol}}$ (initial and final), these were the first two parameters to be selected in this study. The values of the variation of the parameters step$_{\text{ind}}$ (initial and final) and step$_{\text{vol}}$ (initial and final), are presented in Table 5.2. These values were extrapolated from the ones used in the examples presented in [17]. The initial values for these two parameters were considered the total number of possible solutions for the feature selection problem ($2^{60} \approx 1e18$), similar to [17], and the final values were varied.

The combination of the parameters Step$_{\text{ind}}$ and Step$_{\text{vol}}$ ([1e18 1e3] and [1e18 1e5], respectively) were chosen mainly because of the low number of features selected and the better values for the mean ACC of crossvalidation.

It is important to note that, for all combinations presented in Table 5.2, the percentage of contraction, the number of equal barycentre and the plot accounted a low performance of the internal function of the algorithm.
The combination of the parameters Stepind and Stepvol ([1e18 1e3] and [1e18 1e5], respectively) were chosen mainly because of the low number of features selected and the better values for the mean ACC of crossvalidation.

It is important to note that, for all combinations presented in Table 5.2, the percentage of contraction, the number of equal barycentre and the plot accounted a low performance of the internal function of the algorithm.

It was then proceeded the selection of the parameters Wscale and α, Table 5.3. The selected parameters (Wscale=50 and α=0.1) were chosen mainly because of the indicators: percentages of contraction, number of features selected and mean ACC of the crossvalidation values, presented in Table 5.3. The tests using Wscale=50 were the only ones that the contraction to the barycenter did not occurred in all iterations.

Table 5.3: Results of the study the parameters: Wscale and α. Selected set as in bold.

<table>
<thead>
<tr>
<th>Stepind</th>
<th>Stepvol</th>
<th>Wscale</th>
<th>α</th>
<th>Fitness FS</th>
<th>Features selected</th>
<th>ACC FS</th>
<th>Mean ACC crossvalidation</th>
<th>% contraction</th>
<th>No. of same position of barycenter</th>
<th>Plt</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.3</td>
<td>0.79</td>
<td>14</td>
<td>0.87</td>
<td>0.72</td>
<td>0.44</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.5</td>
<td>0.79</td>
<td>17</td>
<td>0.87</td>
<td>0.72</td>
<td>0.4</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.7</td>
<td>0.83</td>
<td>19</td>
<td>0.9</td>
<td>0.75</td>
<td>0.4</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.9</td>
<td>0.9</td>
<td>23</td>
<td>0.93</td>
<td>0.73</td>
<td>0.41</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>1.0</td>
<td>0.91</td>
<td>15</td>
<td>0.9</td>
<td>0.75</td>
<td>0.4</td>
<td>0</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5.2: Results of the study the parameters: stepind and stepvol ([initial final]), selected set at bold

<table>
<thead>
<tr>
<th>Stepind</th>
<th>Stepvol</th>
<th>Wscale</th>
<th>α</th>
<th>Fitness FS</th>
<th>Features selected</th>
<th>ACC FS</th>
<th>Mean ACC crossvalidation</th>
<th>% contraction</th>
<th>No. of same position of barycenter</th>
<th>Plt</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.1</td>
<td>0.78</td>
<td>12</td>
<td>0.63</td>
<td>0.77</td>
<td>1.00</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.3</td>
<td>0.8</td>
<td>14</td>
<td>0.88</td>
<td>0.76</td>
<td>1.00</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.5</td>
<td>0.81</td>
<td>21</td>
<td>0.97</td>
<td>0.77</td>
<td>1.00</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.7</td>
<td>0.85</td>
<td>22</td>
<td>0.94</td>
<td>0.71</td>
<td>1.00</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>[1e18 1e3]</td>
<td>[1e18 1e5]</td>
<td>5000</td>
<td>0.9</td>
<td>0.94</td>
<td>22</td>
<td>0.97</td>
<td>0.73</td>
<td>1.00</td>
<td>0</td>
<td>-</td>
</tr>
</tbody>
</table>
With the set of parameters selected, 100 rounds were simulated, being the results presented in Table 5.4.

Table 5.4: Model assessment results – D2BFSS method using sonar database.

<table>
<thead>
<tr>
<th>Mean</th>
<th>10-fold cross validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 rounds</td>
<td>AUC</td>
</tr>
<tr>
<td>mean</td>
<td>mean</td>
</tr>
<tr>
<td>std</td>
<td>mean</td>
</tr>
<tr>
<td>best round</td>
<td>mean</td>
</tr>
</tbody>
</table>

All tests using D2BFSS presented low convergence (indicator: plot) during its graphical evolution in the FS optimization algorithm. This random dynamic can be visualized in Fig 5.3, which shows the evolution of the best fish per iteration, of the best model of Table 5.4.

![Graphical evolution of the B2DFSS process of feature selection.](image)

Figure 5.3: Graphical evolution of the B2DFSS process of feature selection. Evolution of the fish with best performance per iteration (above) and evolution of the number of features selected of the fish with the best performance per iteration (below).

It can be seen that the number of selected features is always around 20 along the 300 iterations. Convergence is not evident in the graphical evolution of the fitness, ACC and the number of features selected.

5.3.3 Binary Fish School Search

Unlike the D2BFSS algorithm, there were no guidelines to test the parameters range of the BFSS algorithm so, a wider approximation was taken. The tables with the results of the study of parameters are presented in appendix A. Analogously to D2BFSS the first parameters selected were the parameters `thres_c` and the `thresh_v` values. The two structural options were also tested, the use or not of the adaptive threshold in the collective and volitive operators, presented in sections 4.4 and 4.5.
In appendix A, the detailed results of the initial 18 tests are presented, each test corresponding to a different set of the parameters $thres_c$ and the $tresh_v$. These tests used the same partition of samples for FS/MA and train/test sets. The first 9 tests used the non-adaptive approach and others use the adaptive threshold. The variation of parameters for each test is presented in Table 5.5.

Table 5.5: Configuration of the parameters $thres_c$ and $tresh_v$ for each of the 18 tests.

<table>
<thead>
<tr>
<th>Test</th>
<th>$thres_c$</th>
<th>$tresh_v$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>2</td>
<td>0.7</td>
<td>0.7</td>
</tr>
<tr>
<td>3</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>4</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>5</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>6</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>7</td>
<td>0.7</td>
<td>0.7</td>
</tr>
<tr>
<td>8</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>9</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>10</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>11</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>12</td>
<td>0.7</td>
<td>0.7</td>
</tr>
<tr>
<td>13</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>14</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>15</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>16</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>17</td>
<td>0.7</td>
<td>0.7</td>
</tr>
<tr>
<td>18</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Each test considered the variation of the $W_{scale}$ parameter (5, 50, 500, 5000 and 2000), as well as the variation of the parameters $\alpha$ (0.1, 0.3, 0.5, 0.7 and 0.9).

In order to help the visualization of the selection process, Fig. 5.4-5.8 outlines the results for the 18 tests, each colour representing the different combinations of $W_{scale}$ and $\alpha$ to the associated parameters $thres_c$ and $tresh_v$ of each test.

The most discriminate indicators (introduced in section 5.2) that were used to select the best test were: the FS best fitness (Fig. 5.4), the number of features selected (Fig. 5.5), the percentage of contraction of the volitive operator (Fig. 5.6), the number of equal positions of the barycenters (Fig. 5.7), iteration of the optimization algorithm with the best solution (Fig. 5.8), and the plot which can be analysed in the detailed tables in appendix A.

![Graphical representation of the results of tests 1-18 for the indicator: FS best fitness. This indicator indicates the fitness performance of the best model in the FS process, higher the best. The overall best performance for the tests 10-18 is evident.](image)

![Graphical representation of the results of tests 1-18 for the indicator: number of features selected. This indicator indicates the number of features selected in the FS process, lower the best. Tests 10-18 achieved slightly better results for this operator.](image)
After analysing the results from Fig. 5.4-5.8, it was decided that the tests 12 to 16 were the ones with the best configuration. The overall best performance of the tests with the adaptive threshold (12-18) over the ones without it (1-9) was obvious. All the indicators were taken into consideration to these conclusions however, the most incriminating one was the number of repetitions of the same position of the barycentre. Only the test 12-16 performed well from the point of view of this indicator. Recalling that the high number of repeated positions of the barycentre leads to local maxima and, therefore, to a weaker overall performance of the algorithm. In the tables of detailed data in appendix A, it also proved that the indicator plot achieved better performances for the tests 12-16.

After selecting the tests 12-16, it was decided to vary the value of the parameter $W_{\text{scale}}$ to 5, 50, 500, 5000, 20000, 100000 and 1000000, in order to perform a wider analysis of the parameters $W_{\text{scale}}$.

The same strategy here taken to choose the best test. After looking at the detailed data the decisive indicators for selecting the best test and respective set of parameters was the number of repetitions of the same position of the barycentre (Fig. 5.9).
Tests 13 and 16 presented good results in this indicator. However, as shown in the detailed data, the indicator plot of the test 16 proved to be the best. Next, the parameters $W_{\text{scale}}$ and $\alpha$ were selected within the test 16. This was made in a more precise way and looking into the detailed results, concluding $W_{\text{scale}}=100000$. The parameters $W_{\text{scale}}$ proved not to be as relevant as in D2BFSS. The last parameter to be selected was the Stepind initial and final values, the results are shown in Table 5.6.

Table 5.6: Results of the study the parameters: stepind. Selected set in bold.

<table>
<thead>
<tr>
<th>thres_c</th>
<th>thres_v</th>
<th>Stepind</th>
<th>Wscale</th>
<th>$\alpha$</th>
<th>Fitness FS</th>
<th>Features selected</th>
<th>ACC FS</th>
<th>Mean ACC crossvalidation</th>
<th>% contraction</th>
<th>No. of same position of barycenter</th>
<th>Plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>0.3</td>
<td></td>
<td>100000</td>
<td>0.5</td>
<td>0.85</td>
<td>10</td>
<td>87.5</td>
<td>75.83</td>
<td>0.83</td>
<td>59</td>
<td>++</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.5</td>
<td>0.5</td>
<td></td>
<td>0.88</td>
<td>9</td>
<td>90.63</td>
<td>73.85</td>
<td>0.88</td>
<td>91</td>
<td>++</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.5</td>
<td>0.2</td>
<td>0.001</td>
<td>0.89</td>
<td>9</td>
<td>93.75</td>
<td>70.35</td>
<td>0.91</td>
<td>131</td>
<td>++</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.5</td>
<td>0.2</td>
<td>0.001</td>
<td>0.87</td>
<td>12</td>
<td>99.75</td>
<td>74.63</td>
<td>0.89</td>
<td>109</td>
<td>++</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.5</td>
<td>0.01</td>
<td>0.001</td>
<td>0.85</td>
<td>12</td>
<td>90.63</td>
<td>72.99</td>
<td>0.5</td>
<td>26</td>
<td>++</td>
</tr>
</tbody>
</table>

The final configuration (stepind=[0.01 0.001]) of parameters was mainly selected due to the percentage of contraction and the number of the same position of barycentre.

The 100 rounds with the selected set of parameters were then simulated. The results are presented in Table 5.7.

Table 5.7: Model assessment results - BFSS method using sonar database.

<table>
<thead>
<tr>
<th>10-fold cross validation</th>
<th>mean</th>
<th>standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 rounds</td>
<td>AUC</td>
<td>ACC %</td>
</tr>
<tr>
<td>mean</td>
<td>0.72</td>
<td>72.85</td>
</tr>
<tr>
<td>std</td>
<td>0.04</td>
<td>4.27</td>
</tr>
<tr>
<td>best round</td>
<td>0.85</td>
<td>85.62</td>
</tr>
</tbody>
</table>

A plot of the graphical evolution of the FS process using BFSS is shown in Fig. 5.10, presenting the fitness and the accuracy for the fish with best fitness performance per iteration. The number of features selected is also presented.
The plot shows a more explicit convergence throughout the entire process of optimization in comparison with the plot of the D2BFSS algorithm Fig. 5.3.

5.3.4 Binary Particle Swarm Optimization

Analogously to the FSS modified algorithms 300 iterations and 30 particles were used in the BPSO simulations. The BPSO had already been applied to the sonar database in the feature selection problem [64], the parameters suggested in [64] were used:

- $V_{\text{max}} = 5$
- $P_{\text{mut}} = \text{no}$
- $R_{\text{mut}} = 0.5/N_f$
- Reset $x_{\text{sb}}=4$
- $\alpha = 0.7$

The results of the 100 rounds are presented in Table 5.8.

Table 5.8: Model assessment results - BPSO method using sonar database.

<table>
<thead>
<tr>
<th>10-fold cross validation</th>
<th>mean</th>
<th>standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100 rounds</td>
<td>100 rounds</td>
</tr>
<tr>
<td></td>
<td>AUC</td>
<td>ACC %</td>
</tr>
<tr>
<td>mean</td>
<td>0.73</td>
<td>73.3</td>
</tr>
<tr>
<td>std</td>
<td>0.04</td>
<td>4.31</td>
</tr>
<tr>
<td>best round</td>
<td>0.82</td>
<td>82.67</td>
</tr>
</tbody>
</table>
The plot of the FS process for the best round is shown in Fig. 5.11.

![Graphical evolution of the BPSO process of feature selection.](image)

Figure 5.11: Graphical evolution of the BPSO process of feature selection. Evolution of the fish with best performance per iteration (above) and evolution of the number of features selected of the fish with the best performance per iteration (below).

After iteration ~70 the best particle does not change. This behaviour leads us to believe that early convergence to a local maximum could be occurring.

### 5.3.5 Comparison of Feature Selection Methods

It is now possible to compare the performances of the several FS algorithms applied to the sonar database. The results are summarized in Table 5.9. The results of the process described in section 5.2 without using FS (no FS) are also presented. In this case the data was also divided in FS/MA subsets but the FS subset was only used to find the threshold for the data.

Table 5.9- Comparison between the studied FS approaches using sonar database.

<table>
<thead>
<tr>
<th>100 rounds</th>
<th>mean</th>
<th>standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO FS</td>
<td>mean</td>
<td>0.74</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.59</td>
</tr>
<tr>
<td>SFS</td>
<td>mean</td>
<td>72.19</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.04</td>
</tr>
<tr>
<td>BFSS</td>
<td>mean</td>
<td>74.59</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.04</td>
</tr>
<tr>
<td>BPSO</td>
<td>mean</td>
<td>73.3</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.04</td>
</tr>
</tbody>
</table>

The main measures of performance here compared were the mean accuracy (%) of the crossvalidation process and the number of features selected.
As expected, the FS algorithms using metaheuristics optimization algorithms (D2BFSS, BFSS and BPSO) achieved better results in comparison with the SFS and no feature selection approaches. Regarding the number of features selected, the D2BFSS method had lower performance in comparison with the BFSS and the BPSO. This, and the fact that the D2BFSS presented a poor convergence throughout the iterations in the FSS process (section 5.3.2), led to the conclusion that the use of the decimal to binary system was not the best approach to achieve results similar to the state of the art algorithms, reinforcing the need to modify the internal mechanisms of the FSS algorithm to deal with binary input, the BFSS.

It can be concluded that, although the BPSO algorithm had selected 1 less feature in comparison to the BFSS, the performance of BFSS algorithm greatly exceed the BPSO. It is believed that the presence of the collective-volitive operator in the BFSS, which enables the fishes to contract or expand per iteration, is the main tool allowing the algorithm to not converge to local maxima and, thereby, obtaining better results.

5.4-Readmission database

After the processing of the MIMICII database [15], section 2.1.2, there were four different datasets, each dataset including a different gradient of weights for the weighted mean, to characterise the time series for each of the 22 variables considered per patient.

Each dataset is composed of 726 samples (patients) with 132 features. The goal is to classify each of the patients in one of two classes: readmitted or not readmitted within 24 to 72h after discharged. Since the number of samples is quite imbalanced in each class (12.3% readmitted and 87.7% not readmitted) the main performance measure to be maximized is the AUC. Due to the high number of features to be selected and the high number of samples, in comparison to the sonar database, it was chosen to perform 500 rounds of the whole process (FS+MA), always with different partitions of the data for the FS/MA and train/test sets. However, the same partitions between the different FS algorithms were used, allowing the results to be comparable.

For each of the four datasets, the SFS, the BFSS and the BPSO algorithms were applied to the Feature selection. Due to its low overall performance and lack of convergence in the sonar database, the D2BFSS was discarded.
5.4.1 Sequential Forward Selection

The number of features selected in health care problems is crucial. Lower number of features selected corresponds to simpler models, as announced in section 1.1.2. After the analysis of several articles [15,16] that used fuzzy modelling to predict the readmission of ICU patients and feature selection, it was decided to use the stop criterion of the SFS algorithm of 10 features selected. After the stop criteria was met, the set of features selected with the best performance (AUC) was selected.

Table 5.10 shows the results of the 500 rounds for each of the datasets.

Table 5.10: Model assessment results - SFS method using the readmission datasets with different gradients.

<table>
<thead>
<tr>
<th>Gradient</th>
<th>500 rounds</th>
<th>AUC</th>
<th>ACC %</th>
<th>sensitivity</th>
<th>specificity</th>
<th>AUC</th>
<th>ACC %</th>
<th>sensitivity</th>
<th>specificity</th>
<th>threshold</th>
<th>features selected</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>mean</td>
<td>0.63</td>
<td>0.61</td>
<td>0.73</td>
<td>0.53</td>
<td>0.08</td>
<td>17.09</td>
<td>0.23</td>
<td>0.13</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>std</td>
<td>0.02</td>
<td>0.07</td>
<td>0.07</td>
<td>0.01</td>
<td>4.91</td>
<td>0.05</td>
<td>0.05</td>
<td>0.02</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>best round</td>
<td>0.68</td>
<td>64.57</td>
<td>0.73</td>
<td>0.63</td>
<td>19.88</td>
<td>0.13</td>
<td>0.22</td>
<td>0.15</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mean</td>
<td>0.63</td>
<td>0.61</td>
<td>0.72</td>
<td>0.54</td>
<td>17.28</td>
<td>0.23</td>
<td>0.13</td>
<td>0.02</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>std</td>
<td>0.01</td>
<td>0.07</td>
<td>0.07</td>
<td>0.01</td>
<td>4.01</td>
<td>0.05</td>
<td>0.05</td>
<td>0.02</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>best round</td>
<td>0.69</td>
<td>61.28</td>
<td>0.79</td>
<td>0.59</td>
<td>15.59</td>
<td>0.18</td>
<td>0.17</td>
<td>0.15</td>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mean</td>
<td>0.63</td>
<td>0.62</td>
<td>0.71</td>
<td>0.53</td>
<td>17.51</td>
<td>0.23</td>
<td>0.13</td>
<td>0.02</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>std</td>
<td>0.01</td>
<td>0.07</td>
<td>0.07</td>
<td>0.02</td>
<td>4.01</td>
<td>0.05</td>
<td>0.05</td>
<td>0.02</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>best round</td>
<td>0.68</td>
<td>61.37</td>
<td>0.91</td>
<td>0.45</td>
<td>9.35</td>
<td>0.11</td>
<td>0.10</td>
<td>0.10</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mean</td>
<td>0.63</td>
<td>0.67</td>
<td>0.72</td>
<td>0.54</td>
<td>17.52</td>
<td>0.23</td>
<td>0.13</td>
<td>0.02</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>std</td>
<td>0.02</td>
<td>0.07</td>
<td>0.07</td>
<td>0.02</td>
<td>4.14</td>
<td>0.05</td>
<td>0.05</td>
<td>0.02</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>best round</td>
<td>0.68</td>
<td>66.43</td>
<td>0.71</td>
<td>0.65</td>
<td>14.59</td>
<td>0.14</td>
<td>0.16</td>
<td>0.15</td>
<td>9</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The results show that the use of different gradients to the weights of the weighted mean was not significant. The number of features selected and the corresponded AUC measure did not allow the retrieval of a definitive pattern of the best gradient.

5.4.2 Binary Fish School Search

The stop criterion selected to the metaheuristic algorithms (BFSS and BPSO) was the number of iterations, 500 iterations with 10 (fish/particles). This selection was made taking into consideration the max number of features to be selected.

Once again, a study was performed to select the parameters. The first parameters to be selected were the `thres_c` and `thres_v`. The conclusions of the study of parameters made to the sonar database were here used, only the 5 tests with significant better overall performance were here used to select `thres_c` and `thres_v` parameters. The configuration of parameters in this 5 tests is presented in Table 5.11.

Table 5.11: Configuration of the parameters `thres_c` and `thres_v` for each of the 5 tests.

<table>
<thead>
<tr>
<th>Test</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thres_c</td>
<td>0.5</td>
<td>0.3</td>
<td>0.1</td>
<td>0.9</td>
<td>0.7</td>
</tr>
<tr>
<td>Thres_v</td>
<td>0.5</td>
<td>0.3</td>
<td>0.1</td>
<td>0.1</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Adaptative threshold

56
These combinations of the parameters correspond to the best 5 configurations in the first study of the parameters of the sonar database, (tests 12, 13, 14, 15 and 16 in appendix A, with the adaptive threshold). The results of these 5 configurations, presented in section 5.3.3, were so evident that there was no need to test again all the configurations.

As the four datasets only differ in the 22 features of the weighted mean, it was decided to do the study of parameters for only one dataset. The detailed results for these 5 tests are attached in appendix B.

The same strategy as in the sonar database was adopted for choosing the best test. For each test, the parameters $W_{\text{scale}}$ (values of 5, 50, 500, 5000, 20000, 100000 and 1000000) and $\alpha$ (values of 0.1, 0.3, 0.5, 0.7 and 0.9) were tested.

After the observation of the detailed results, it was evident that the indicator that stood out was the plot. The other operators prove not to be conclusive As an example, Fig 5.12 illustrates the graphical representation of three indicators.

As the graphical visualization is a qualitative analyse and its goal is to exclude tests that have obvious weak performance, it was not possible to conclude which was the test with better performance. However, as shown in the appendix B, the plot indicator states that test 3 has a really good convergence in the visualization of the graphical evolution in comparison with the other tests.

Figure 5.11: Graphical representation of the results of tests 1-5.
The selection of the parameters $W_{\text{scale}}$ and $\alpha$ was made observing the detailed data from test 3, $W_{\text{scale}}=500$ and $\alpha=0.1$. The measures that had major influence in this choice were the number of features selected (low) and the performance of the best model in the FS process (high). The final parameter to be optimized was the parameter StepInd. The results are presented in Table 5.12.

The final value of the StepInd of [0.01 0.001] was selected mainly because of the percentage of contraction. Reminding that very low percentage of contraction in the volitive operator means no convergence, low exploitation, and very high percentage means convergence to local maxima, low exploration.

With a viable set of parameters selected, the 500 rounds were performed for the 4 datasets with different gradients for the weights of the weighted mean, being the results presented in Table 5.13.

As occurred in the SFS results, the results of the usage of different gradients for the weighted means prove that there was not a gradient that achieved significantly better performance. In title of example, the graphical evolution of the FS process using BFSS and the dataset with gradient of 0.1 is presented in Fig. 5.13.

The graphical evolution in Fig 5.13, unquestionably confirms the convergence of the novel BFSS algorithm formulated in this thesis. In the first ~200 iterations there is a transitory dynamic, where the BFSS algorithm mainly converges to a lower number of features. In the rest of the iterations, the algorithm searches for the optimal combination of a low number of features and a high performance of AUC.
It is important to recall that the BFSS’s internal mechanisms, more specifically the collective-volitive operator imposes a contraction (if the individual search is successfully) or an expansion (if the individual search is unsuccessfully) which results in the oscillation of the graphical evolution. This effect is highly beneficial in terms of avoiding convergence to local maxima, and still maintaining the convergence.

5.4.3 Binary Particle Swarm Optimization

The BPSO had already been used in feature selection problems using datasets derived from the MIMICII database. Even though the Shannon entropy and the weighted mean were introduced, the overall characteristics of the database are very similar to those datasets used before [17]. Thus, it was decided to use the parameters referred in [17] with the exception of the parameter $\alpha$:

- $V_{\text{max}} = 5$
- $P_{\text{mut}} = \text{no}$
- $R_{\text{mut}} = 0.5/N_{f}$
- $\text{Reset } x_{sb} = 4$

With the introduction of the Shannon entropy and weighted mean, the number of total features to be selected was higher than in [17]. The parameter $\alpha$ is directly related to the final number
of features selected and is affected by the total number of features. It was necessary to do an analysis to select the value of the parameter $\alpha$. Table 5.14 shows the results of the tests. Once again, this analysis was made only for the dataset with the gradient 0.1.

Table 5.14: Results of the study the parameters: $\alpha$. Selected in bold.

<table>
<thead>
<tr>
<th>Study of the parameter: $\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
</tr>
<tr>
<td>0.1</td>
</tr>
<tr>
<td>0.3</td>
</tr>
<tr>
<td>0.5</td>
</tr>
<tr>
<td>0.7</td>
</tr>
<tr>
<td>0.9</td>
</tr>
</tbody>
</table>

It was decided to use the $\alpha$ value of 0.5. The final set of parameters were then used to perform the 500 rounds to each of the 4 readmission datasets, the results are presented in Table 5.15.

Table 5.15: Model assessment results - BPSO method using the readmission datasets with different gradients.

<table>
<thead>
<tr>
<th>Gradient</th>
<th>500 rounds</th>
<th>AUC</th>
<th>ACC %</th>
<th>sensitivity</th>
<th>specificity</th>
<th>AUC</th>
<th>ACC %</th>
<th>sensitivity</th>
<th>specificity</th>
<th>threshold</th>
<th>features selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>mean</td>
<td>0.62</td>
<td>53.60</td>
<td>0.73</td>
<td>0.51</td>
<td>0.07</td>
<td>17.53</td>
<td>0.24</td>
<td>0.23</td>
<td>0.14</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.02</td>
<td>7.18</td>
<td>0.09</td>
<td>0.08</td>
<td>0.02</td>
<td>5.50</td>
<td>0.07</td>
<td>0.07</td>
<td>0.03</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>best round</td>
<td>0.68</td>
<td>56.72</td>
<td>0.83</td>
<td>0.53</td>
<td>0.08</td>
<td>15.92</td>
<td>0.16</td>
<td>0.19</td>
<td>0.15</td>
<td>11</td>
</tr>
<tr>
<td>0.4</td>
<td>mean</td>
<td>0.53</td>
<td>68.54</td>
<td>0.33</td>
<td>0.75</td>
<td>0.11</td>
<td>7.91</td>
<td>0.20</td>
<td>0.09</td>
<td>0.15</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.04</td>
<td>7.05</td>
<td>0.14</td>
<td>0.15</td>
<td>0.03</td>
<td>2.25</td>
<td>0.06</td>
<td>0.02</td>
<td>0.02</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>best round</td>
<td>0.67</td>
<td>68.89</td>
<td>0.64</td>
<td>0.70</td>
<td>0.15</td>
<td>8.58</td>
<td>0.25</td>
<td>0.08</td>
<td>0.15</td>
<td>11</td>
</tr>
<tr>
<td>0.6</td>
<td>mean</td>
<td>0.62</td>
<td>53.80</td>
<td>0.73</td>
<td>0.51</td>
<td>0.07</td>
<td>17.85</td>
<td>0.25</td>
<td>0.23</td>
<td>0.14</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.02</td>
<td>6.51</td>
<td>0.08</td>
<td>0.08</td>
<td>0.02</td>
<td>5.53</td>
<td>0.07</td>
<td>0.07</td>
<td>0.03</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>best round</td>
<td>0.69</td>
<td>58.44</td>
<td>0.84</td>
<td>0.55</td>
<td>0.12</td>
<td>14.69</td>
<td>0.17</td>
<td>0.16</td>
<td>0.15</td>
<td>14</td>
</tr>
<tr>
<td>0.9</td>
<td>mean</td>
<td>0.59</td>
<td>57.66</td>
<td>0.62</td>
<td>0.57</td>
<td>0.08</td>
<td>14.69</td>
<td>0.23</td>
<td>0.18</td>
<td>0.14</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.05</td>
<td>9.75</td>
<td>0.23</td>
<td>0.14</td>
<td>0.02</td>
<td>4.58</td>
<td>0.07</td>
<td>0.09</td>
<td>0.03</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>best round</td>
<td>0.67</td>
<td>59.13</td>
<td>0.78</td>
<td>0.57</td>
<td>0.12</td>
<td>10.23</td>
<td>0.31</td>
<td>0.14</td>
<td>0.10</td>
<td>8</td>
</tr>
</tbody>
</table>

The results of the usage of the BPSO algorithm in the features selection process show that the gradient of 0.9 achieved better overall performance than the other gradients datasets, mainly because of the low number of features selected. Nonetheless, it is important to note that the results are not highly evident, reinforcing the idea that the use of different gradients is not significant. As an illustrative example, the graphical evolution of the best model using gradient 0.9 is presented in Fig. 5.14. The graphic evolution shows an explicit convergence of the BPSO algorithm. Unlike the BFSS algorithm, after the transient dynamic (1st to ~250th iteration), the oscillation of the best particle in each iteration is not sharp. This fact can be related to a lack of exploration and consequent lower performance. In the author’s opinion, this is the main disadvantage of using the BPSO over the BFSS.
For each of the 4 datasets, the whole process was computed without using a FS algorithm (using all available features). The results of the 500 rounds are presented in Table 5.16.

Table 5.16: Model assessment results – without feature selection using the readmission datasets with different gradients.

<table>
<thead>
<tr>
<th>Gradient</th>
<th>500 rounds</th>
<th>AUC</th>
<th>ACC %</th>
<th>sensitivity</th>
<th>specificity</th>
<th>AUC</th>
<th>ACC %</th>
<th>sensitivity</th>
<th>specificity</th>
<th>threshold</th>
<th>features selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>mean</td>
<td>0.59</td>
<td>48.78</td>
<td>0.72</td>
<td>0.46</td>
<td>0.06</td>
<td>28.10</td>
<td>0.34</td>
<td>0.37</td>
<td>0.06</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.02</td>
<td>10.24</td>
<td>0.12</td>
<td>0.13</td>
<td>0.02</td>
<td>0.07</td>
<td>0.07</td>
<td>0.07</td>
<td>0.02</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>best round</td>
<td>0.65</td>
<td>64.84</td>
<td>0.66</td>
<td>0.65</td>
<td>0.08</td>
<td>19.88</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>132</td>
</tr>
<tr>
<td>0.4</td>
<td>mean</td>
<td>0.59</td>
<td>49.31</td>
<td>0.71</td>
<td>0.46</td>
<td>0.06</td>
<td>28.49</td>
<td>0.34</td>
<td>0.37</td>
<td>0.31</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.02</td>
<td>10.39</td>
<td>0.11</td>
<td>0.13</td>
<td>0.02</td>
<td>0.05</td>
<td>0.07</td>
<td>0.07</td>
<td>0.05</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>best round</td>
<td>0.66</td>
<td>66.89</td>
<td>0.66</td>
<td>0.67</td>
<td>0.11</td>
<td>0.25</td>
<td>0.27</td>
<td>0.32</td>
<td>0.25</td>
<td>132</td>
</tr>
<tr>
<td>0.6</td>
<td>mean</td>
<td>0.59</td>
<td>49.84</td>
<td>0.70</td>
<td>0.47</td>
<td>0.06</td>
<td>28.69</td>
<td>0.34</td>
<td>0.37</td>
<td>0.31</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.02</td>
<td>9.73</td>
<td>0.11</td>
<td>0.13</td>
<td>0.02</td>
<td>0.06</td>
<td>0.06</td>
<td>0.06</td>
<td>0.05</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>best round</td>
<td>0.66</td>
<td>60.59</td>
<td>0.74</td>
<td>0.59</td>
<td>0.08</td>
<td>17.77</td>
<td>0.14</td>
<td>0.21</td>
<td>0.20</td>
<td>132</td>
</tr>
<tr>
<td>0.9</td>
<td>mean</td>
<td>0.58</td>
<td>48.77</td>
<td>0.71</td>
<td>0.46</td>
<td>0.06</td>
<td>28.45</td>
<td>0.34</td>
<td>0.37</td>
<td>0.32</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.02</td>
<td>9.80</td>
<td>0.11</td>
<td>0.13</td>
<td>0.02</td>
<td>0.06</td>
<td>0.06</td>
<td>0.06</td>
<td>0.05</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>best round</td>
<td>0.65</td>
<td>67.58</td>
<td>0.63</td>
<td>0.68</td>
<td>0.10</td>
<td>11.73</td>
<td>0.21</td>
<td>0.14</td>
<td>0.20</td>
<td>132</td>
</tr>
</tbody>
</table>

The results show that with no feature selection, the performances using the different datasets are very similar.
5.4.4 Discuss

After collecting the results of all the methods of feature selection for the readmission datasets, it can be affirmed that the use of different gradients for the weights in the weighted mean proved to be weakly relevant. With the exception of the results using the BPSO algorithm, all the algorithms show almost no sensitivity to the presence of different gradients for the linear distribution of the weights for the weighted mean for the 22 physiologic variables.

The performance of the different feature selection algorithms can be compared using the same dataset. It was decided to collect the results of one dataset to do the comparison, Table 5.17 summarizes the results.

Table 5.17: Comparison between the studied FS approaches using the readmission dataset with gradient:0.9.

<table>
<thead>
<tr>
<th>NO FS</th>
<th>mean, 10-fold cross validation</th>
<th>standard deviation, 10-fold cross validation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC</td>
<td>ACC %</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>std</td>
<td>0.02</td>
<td>9.80</td>
</tr>
<tr>
<td>best round</td>
<td>0.65</td>
<td>67.58</td>
</tr>
<tr>
<td>SFS</td>
<td>mean</td>
<td>0.64</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.02</td>
</tr>
<tr>
<td>best round</td>
<td>0.68</td>
<td>66.43</td>
</tr>
<tr>
<td>BFSS</td>
<td>mean</td>
<td>0.61</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.03</td>
</tr>
<tr>
<td>best round</td>
<td>0.69</td>
<td>55.53</td>
</tr>
<tr>
<td>BPSO</td>
<td>mean</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.05</td>
</tr>
<tr>
<td>best round</td>
<td>0.67</td>
<td>59.13</td>
</tr>
</tbody>
</table>

The results of the comparison of the different features algorithms are not as obvious as the ones with the benchmark sonar database were, possibly due to the variability introduced dealing with real data in high dimensions spaces. The overall performance of the metaheuristic algorithms achieved better results in comparison with the SFS and no feature selection algorithm results.

Once again, the BFSS achieved better overall results in comparison with all methods used, achieving slightly better mean of AUC and less features selected.

It is noteworthy to mention that in all of the datasets used, the BFSS (Table 5.13) achieved considerably less features than all the other methods, maintaining the convergence of the algorithm and slightly better performance of the models.

It is also relevant to note that the sensitivity of the best model using the BFSS algorithm is considerably better in comparison with that of other algorithms. Recalling that this measure indicates the true positive rate, i.e. the patients who were selected as readmitted and were in fact.
Chapter 6

Conclusion

This work has addressed the problem of feature selection. Firstly, the background for machine learning was presented, followed by the definition of its application to the problem of classification. Then, the two databases to be used were introduced, consisting of the benchmark sonar database and the readmission datasets arising from the MIMIC II database. Further along, the fuzzy classification models and C-mean clustering techniques were briefly described.

The modifications that allowed the optimization algorithm Fish School Search to be used in binary input problems were formulated and then applied to the feature selection problem, emphasizing the novel Binary Fish school Search. Other state of the art algorithms for FS were also described, namely the SFS and the BPSO approaches.

Finally, the proposed methods were validated over the benchmark sonar database and then applied to a case of study: the prediction of patient’s readmission in an ICU within 24-72h period following their discharge.

This chapter summarizes the conclusions acquired during the development of this work, and suggests topics for future research.

6.1- Binary Fish School Search

In general, the modification of an existing algorithm is risky. In advance, it is very difficult to the algorithm developer to guarantee that the algorithm will have a better performance that the state of the art algorithms or even to assure convergence.

There are countless ways to modify algorithms with real encoding schemes in order to solve binary input problems. In this thesis, two novel approaches to the FSS algorithm were presented, the D2BFSS and the BFSS. The decimal to binary approach was achieved through simple passage from continuous to discrete system in the usage of the objective function to the original FSS algorithm. The BFSS was a more complex approach that modified the internal mechanisms of the FSS algorithm, allowing the procedure itself to manipulate the binary inputs.
Results of the benchmark sonar database for the feature selection problem showed that simple manipulation of the fitness function, in order to transform the objective function from decimal to binary system, achieved low performance, as well as a very low convergence evolution, when compared to the BFSS algorithm. This result was expected due to the simplicity of the D2BFSS, and reinforced the need to implement internal changes in FSS algorithm.

The results of the two databases tested showed that the initial development goals proposed for the BFSS algorithm were achieved: to remain faithful to the original internal mechanisms of the FSS without losing the meaning of each operator, addiction of few number of parameters, overall simplicity and understanding of the modifications, and, mainly, the convergence of the algorithm.

However, it is important to note that the number of parameters used in the BFSS process is high in comparison with other Feature Selection algorithms, and some limitations to the proposed algorithm can arise if there is a poor adjustment in the parameters.

To conclusion, this brand new binary optimization algorithm exceeded expectations, with not only its convergent evolution but also with good results in comparison with the other features selection algorithms, especially the BPSO. The main contribution to achieve such results is believed to have been the presence of the collective-volitive operator, which had a major role in the exploration process of the algorithm, and consequent capacity of avoiding local maxima.

It is also important to note that although the formulated BFSS algorithm was used in the problem of feature selection, it can be applied to any optimization problem with a binary encoding of the inputs, opening doors for future fields of research.

6.2- Prediction of readmissions

The dimensionality of medical data is typically very high. Hence the great importance of using algorithms of feature selection for this environment, improving early detection of medical problems. The proposed wrapper methods are appropriated to this problem due to various reasons, such as the possibility to deal with large databases, the capacity to predict outcomes by means of a small subset of features and the possibility to bring out new set of variables never considered before.

However, the proposal of using different gradients for the linear distribution of the weights in the weighted mean to describe the time series of the physiologic variables, proved not to be significant. It was expected that one gradient for the weights of the weighted mean would stand out, the results show that the use of different gradients did not present a significant benefit in the performance of the models nor in the number of Features Selected.
6.3- Future work

In this work, various modifications to the FSS algorithm were proposed. Nevertheless, not all the possible paths were explored. The most promising future research topics in the BPSO algorithm could be the following:

- **Use of the BFSS algorithm in other application**: although the presented BFSS algorithm was used in the feature selection problem, it can be used for any optimization problem with a binary encoding of the inputs.

- **Reduction of the number of parameters**: lower number of parameters means less time in their selection and, generally, more simplicity. This research topic must be done maintaining the good performance of the models and low number of features selected.

- **Use of newly different parameters update strategies**: instead of using constant parameters in all iterations of the algorithm, using varying values over the iterations. Studies [31] already proved that the use of dynamic values for the weights of the fishes in FSS algorithms can be beneficial.

- **Development and refinement of FSS operators**: testing other possible paths in the encoding of the binary inputs.

To prove the full potential of the BFSS algorithm it is of major important to:

- **Validate de results in other benchmark databases**: testing for databases with different dimensions and compare the results with other state of the art algorithms.

- **Not using the weighted mean in the MIMIC II database**: using the same kind of statistical measures as the latest studies in the prediction of readmission of ICU patients. This would allow the comparison of the performance of the BFSS to the state of the art features selection algorithms in the readmission problem.
Bibliography


Appendix
A  Extended Results - sonar database

In order to choose an appropriated set of parameters to the optimization algorithm, it is important to do a coherent study to the parameters to be used. Some limitations can arise if there is a poor adjustment in the parameters. With this in mind, we present the detailed results of the 16 tests used in the study of the parameters thres_c and thres_v, wscale and α, for the sonar database. The 1-9 tests used the non-adaptative threshold and the 10 -18 used the adaptive threshold.

Table A.1: Results of the study the parameters: thres_c and thres_v - Tests 1-3

| Test No. | thres_c | thres_v | Bipolar | Wscale | Wscale | Stepind | Stepvol | Sensitivity with the best method [AVG], ACC FS, MeanAUC, % contrasts, % of cases classification, IPmat |
|----------|---------|---------|---------|--------|--------|---------|---------|------------------------------------------------|--------------------------------------------------|--------------------------------------------------|
| 1        | 0.7     | 0.6     | 1.0     | 0.9    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 2        | 0.6     | 0.6     | 0.8     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 3        | 0.5     | 0.6     | 0.3     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 4        | 0.4     | 0.6     | 0.1     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 5        | 0.3     | 0.6     | 0.0     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 6        | 0.2     | 0.6     | -       | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 7        | 0.1     | 0.6     | -       | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 8        | 0.0     | 0.6     | -       | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 9        | 0.9     | 0.6     | 0.9     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 10       | 0.8     | 0.6     | 0.8     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 11       | 0.7     | 0.6     | 0.7     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 12       | 0.6     | 0.6     | 0.6     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 13       | 0.5     | 0.6     | 0.5     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 14       | 0.4     | 0.6     | 0.4     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 15       | 0.3     | 0.6     | 0.3     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 16       | 0.2     | 0.6     | 0.2     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 17       | 0.1     | 0.6     | 0.1     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |
| 18       | 0.0     | 0.6     | 0.0     | 0.5    | 0.5    | 0.1     | 0.1     | 0.78, 0.81, 0.55, 0.64, 61.31, 64.00, 0.67, 289 |

Note: The table shows the results of different tests with varying parameters. Each test is identified by a number, and the parameters are adjusted accordingly. The results include the accuracy (ACC FS) and the mean AUC (MeanAUC) for each test.
Table A.2: Results of the study the parameters: thres_c and thres_v - Tests 4-7

<table>
<thead>
<tr>
<th>Test No.</th>
<th>thres_c</th>
<th>thres_v</th>
<th>Steepgrad</th>
<th>Unclele</th>
<th>n</th>
<th>Fitness FS</th>
<th>Features selected</th>
<th>Iteration with the best model in FS</th>
<th>ACC FS</th>
<th>MeanAUC</th>
<th>% iteration</th>
<th>No. of runs with the best model in FS</th>
<th>AMP baricenter</th>
<th>Plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>0.9</td>
<td>0.4</td>
<td>[0.2 0.01]</td>
<td>50</td>
<td>7</td>
<td>0.94</td>
<td>22</td>
<td>61.39</td>
<td>0.92</td>
<td>0.92</td>
<td>0.92</td>
<td>2000</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.4</td>
<td>0.1</td>
<td>[0.2 0.01]</td>
<td>50</td>
<td>7</td>
<td>0.94</td>
<td>22</td>
<td>61.39</td>
<td>0.92</td>
<td>0.92</td>
<td>0.92</td>
<td>2000</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.8</td>
<td>0.1</td>
<td>[0.2 0.01]</td>
<td>50</td>
<td>7</td>
<td>0.94</td>
<td>22</td>
<td>61.39</td>
<td>0.92</td>
<td>0.92</td>
<td>0.92</td>
<td>2000</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.7</td>
<td>0.3</td>
<td>[0.2 0.01]</td>
<td>50</td>
<td>7</td>
<td>0.94</td>
<td>22</td>
<td>61.39</td>
<td>0.92</td>
<td>0.92</td>
<td>0.92</td>
<td>2000</td>
<td>50</td>
<td></td>
</tr>
</tbody>
</table>

Study of the parameter: thres_c and thres_v (initial and final value), wscale and α

Crossvalidation contraction No. of same baricenter
<table>
<thead>
<tr>
<th>Test No.</th>
<th>thres_c</th>
<th>thres_v</th>
<th>Stepind</th>
<th>Wscale</th>
<th>α</th>
<th>Fitness FS</th>
<th>Features selected</th>
<th>Iteration with the best model in FS</th>
<th>ACC FS</th>
<th>Mean AUC</th>
<th>% contraction</th>
<th>No. of runs</th>
<th>POSITION of balicenter</th>
<th>Plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>0.2</td>
<td>0.6</td>
<td>0.2</td>
<td>0.01</td>
<td></td>
<td>0.04</td>
<td>0.02</td>
<td>0.02</td>
<td>0.04</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>9</td>
<td>0.3</td>
<td>0.7</td>
<td>0.2</td>
<td>0.01</td>
<td></td>
<td>0.04</td>
<td>0.02</td>
<td>0.02</td>
<td>0.04</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>10</td>
<td>0.8</td>
<td>0.8</td>
<td>0.2</td>
<td>0.01</td>
<td></td>
<td>0.04</td>
<td>0.02</td>
<td>0.02</td>
<td>0.04</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>11</td>
<td>0.7</td>
<td>0.7</td>
<td>0.2</td>
<td>0.01</td>
<td></td>
<td>0.04</td>
<td>0.02</td>
<td>0.02</td>
<td>0.04</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
<td>0.02</td>
</tr>
</tbody>
</table>

Table A.3: Results of the study the parameters: thres_c and thres_v - Tests 8-11.
<table>
<thead>
<tr>
<th>Test No.</th>
<th>thres c</th>
<th>thres v</th>
<th>Seeded</th>
<th>Unseeded</th>
<th>thres c</th>
<th>thres v</th>
<th>Seeded</th>
<th>Unseeded</th>
<th>thres c</th>
<th>thres v</th>
<th>Seeded</th>
<th>Unseeded</th>
<th>thres c</th>
<th>thres v</th>
<th>Seeded</th>
<th>Unseeded</th>
<th>thres c</th>
<th>thres v</th>
<th>Seeded</th>
<th>Unseeded</th>
<th>thres c</th>
<th>thres v</th>
<th>Seeded</th>
<th>Unseeded</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.0</td>
<td>0.05</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td>0.9</td>
<td>0.1</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.0</td>
<td>0.05</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td>0.9</td>
<td>0.1</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>0.0</td>
<td>0.05</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td>0.9</td>
<td>0.1</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>0.0</td>
<td>0.05</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td>0.9</td>
<td>0.1</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>0.0</td>
<td>0.05</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td>0.9</td>
<td>0.1</td>
<td>1</td>
<td>500</td>
<td>63.92</td>
<td>59.91</td>
<td>0.80</td>
<td>112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table A.4: Results of the study parameters: thres c and thres v - Tests 12-14**
<table>
<thead>
<tr>
<th>Test No.</th>
<th>thre_c</th>
<th>thre_v</th>
<th>Stepind</th>
<th>Wscale</th>
<th>90%</th>
<th>60%</th>
<th>30%</th>
<th>% contraction</th>
<th>No. of same position of hologram</th>
<th>Plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.95</td>
<td>1</td>
<td>105</td>
<td>81.06</td>
<td>72.17</td>
<td>0.59</td>
<td>32</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.95</td>
<td>1</td>
<td>106</td>
<td>79.81</td>
<td>78.63</td>
<td>0.93</td>
<td>32</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0.95</td>
<td>2</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.95</td>
<td>3</td>
<td>74</td>
<td>64.53</td>
<td>67.74</td>
<td>0.92</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>0.95</td>
<td>4</td>
<td>72</td>
<td>64.53</td>
<td>67.74</td>
<td>0.92</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.96</td>
<td>5</td>
<td>50</td>
<td>93.75</td>
<td>79.64</td>
<td>0.96</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.96</td>
<td>5</td>
<td>50</td>
<td>93.75</td>
<td>79.64</td>
<td>0.96</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0.96</td>
<td>5</td>
<td>50</td>
<td>93.75</td>
<td>79.64</td>
<td>0.96</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.96</td>
<td>5</td>
<td>50</td>
<td>93.75</td>
<td>79.64</td>
<td>0.96</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>0.96</td>
<td>5</td>
<td>50</td>
<td>93.75</td>
<td>79.64</td>
<td>0.96</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.95</td>
<td>6</td>
<td>79</td>
<td>91.05</td>
<td>74.69</td>
<td>0.98</td>
<td>38</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.95</td>
<td>6</td>
<td>79</td>
<td>91.05</td>
<td>74.69</td>
<td>0.98</td>
<td>38</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0.95</td>
<td>6</td>
<td>79</td>
<td>91.05</td>
<td>74.69</td>
<td>0.98</td>
<td>38</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.95</td>
<td>6</td>
<td>79</td>
<td>91.05</td>
<td>74.69</td>
<td>0.98</td>
<td>38</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>0.95</td>
<td>6</td>
<td>79</td>
<td>91.05</td>
<td>74.69</td>
<td>0.98</td>
<td>38</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.95</td>
<td>7</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.95</td>
<td>7</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0.95</td>
<td>7</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.95</td>
<td>7</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>0.95</td>
<td>7</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.95</td>
<td>8</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.95</td>
<td>8</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0.95</td>
<td>8</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.95</td>
<td>8</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>0.95</td>
<td>8</td>
<td>82</td>
<td>63.75</td>
<td>70.74</td>
<td>0.96</td>
<td>257</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table A.5: Results of the study of the parameters: thre_c and thres_v - Tests 15-17
Table A.6: Results of the study the parameters: thres_c and thres_v – Test 18

<table>
<thead>
<tr>
<th>Test No.</th>
<th>thres_c</th>
<th>thres_v</th>
<th>Stepind</th>
<th>Wscale</th>
<th>α</th>
<th>Fitness FS</th>
<th>Features selected</th>
<th>Iteration with the best model in FS</th>
<th>ACC FS</th>
<th>MeanAUC crossvalidation</th>
<th>% contraction</th>
<th>No. of same position of parameter</th>
<th>Plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.1</td>
<td>0.96</td>
<td>1</td>
<td>263</td>
<td>75.13</td>
<td>65.86</td>
<td>0.40</td>
<td>187</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.1</td>
<td>0.97</td>
<td>1</td>
<td>90</td>
<td>75.13</td>
<td>65.85</td>
<td>0.46</td>
<td>234</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0.96</td>
<td>1</td>
<td>106</td>
<td>75.13</td>
<td>65.85</td>
<td>0.46</td>
<td>234</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.91</td>
<td>6</td>
<td>588</td>
<td>80.68</td>
<td>68.79</td>
<td>0.50</td>
<td>252</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.91</td>
<td>1</td>
<td>189</td>
<td>78.37</td>
<td>69.96</td>
<td>0.49</td>
<td>185</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.91</td>
<td>2</td>
<td>126</td>
<td>81.71</td>
<td>69.22</td>
<td>0.46</td>
<td>278</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>0.91</td>
<td>2</td>
<td>53</td>
<td>81.71</td>
<td>72.61</td>
<td>0.52</td>
<td>233</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.91</td>
<td>6</td>
<td>514</td>
<td>70.68</td>
<td>73.84</td>
<td>0.56</td>
<td>242</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.96</td>
<td>2</td>
<td>176</td>
<td>75.13</td>
<td>69.96</td>
<td>0.49</td>
<td>185</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.91</td>
<td>7</td>
<td>117</td>
<td>81.71</td>
<td>73.84</td>
<td>0.56</td>
<td>242</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.93</td>
<td>5</td>
<td>187</td>
<td>87.56</td>
<td>68.24</td>
<td>0.55</td>
<td>250</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.96</td>
<td>5</td>
<td>132</td>
<td>87.56</td>
<td>69.06</td>
<td>0.58</td>
<td>248</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.93</td>
<td>8</td>
<td>185</td>
<td>87.56</td>
<td>71.84</td>
<td>0.50</td>
<td>235</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.94</td>
<td>5</td>
<td>187</td>
<td>76.13</td>
<td>68.08</td>
<td>0.51</td>
<td>183</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.92</td>
<td>8</td>
<td>130</td>
<td>87.56</td>
<td>71.74</td>
<td>0.50</td>
<td>233</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.92</td>
<td>6</td>
<td>187</td>
<td>87.56</td>
<td>68.08</td>
<td>0.51</td>
<td>183</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.92</td>
<td>11</td>
<td>106</td>
<td>87.56</td>
<td>71.74</td>
<td>0.50</td>
<td>233</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.95</td>
<td>1</td>
<td>226</td>
<td>88.75</td>
<td>69.70</td>
<td>0.38</td>
<td>126</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.95</td>
<td>2</td>
<td>80</td>
<td>88.75</td>
<td>71.70</td>
<td>0.38</td>
<td>233</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.95</td>
<td>8</td>
<td>73</td>
<td>88.75</td>
<td>67.17</td>
<td>0.39</td>
<td>269</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0.93</td>
<td>6</td>
<td>92</td>
<td>87.56</td>
<td>71.84</td>
<td>0.50</td>
<td>242</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.95</td>
<td>15</td>
<td>97</td>
<td>86.88</td>
<td>67.68</td>
<td>0.50</td>
<td>235</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
We present the detailed results of the 5 tests used in the study of the parameters \( \text{thres}_c \) and \( \text{thres}_v \), \( \text{wscale} \) and \( \alpha \) to the readmission datasets. The 5 tests used the adaptative threshold.

Table B.1: Results of the study the parameters: \( \text{thres}_c \) and \( \text{thres}_v \) - Tests 1-2

<table>
<thead>
<tr>
<th>Test No.</th>
<th>thre_c</th>
<th>thre_v</th>
<th>Stepind</th>
<th>Wscale</th>
<th>( \alpha )</th>
<th>Fitness FS</th>
<th>Feature selected</th>
<th>MeanAUC</th>
<th>% contraction</th>
<th>No of same position of baricenter</th>
<th>Plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.2</td>
<td>0.01</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0.5</td>
<td>0.2</td>
<td>0.01</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td></td>
</tr>
</tbody>
</table>
Table B.2: Results of the study the parameters: \( \text{thres}_c \) and \( \text{thres}_v \) - Tests 3-5

<table>
<thead>
<tr>
<th>Test No</th>
<th>( \text{thres}_c )</th>
<th>( \text{thres}_v )</th>
<th>Iteration with the best model in FS</th>
<th>ACC FS</th>
<th>MeanAUC crossvalidation</th>
<th>% contraction</th>
<th>No. of same position of the parameter: stepind e stepvol [initial and final value], wscale and ( \alpha )</th>
<th>Plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1 0.9 0.9 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5</td>
<td>0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49</td>
<td>0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59</td>
<td>0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59</td>
<td>3000</td>
</tr>
<tr>
<td>2</td>
<td>0.1 0.9 0.9 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5</td>
<td>0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49</td>
<td>0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59</td>
<td>0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59</td>
<td>15000</td>
</tr>
<tr>
<td>3</td>
<td>0.1 0.9 0.9 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5</td>
<td>0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49</td>
<td>0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59</td>
<td>0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59</td>
<td>50000</td>
</tr>
<tr>
<td>4</td>
<td>0.1 0.9 0.9 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1</td>
<td>0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5</td>
<td>0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49 0.49</td>
<td>0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59</td>
<td>0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59 0.59</td>
<td>100000</td>
</tr>
</tbody>
</table>

Study of the parameter: meanAUC crossvalidation (initial and final value)