UTADIS BASED MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS FOR MEDICAL DIAGNOSIS PROBLEMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

HALENUR ŞAHİN MAHMUTOĞULLARI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
INDUSTRIAL ENGINEERING

DECEMBER 2019

Approval of the thesis:

UTADIS BASED MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS FOR MEDICAL DIAGNOSIS PROBLEMS

submitted by **HALENUR ŞAHİN MAHMUTOĞULLARI** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Yaşar Yasemin Serin
Head of Department, **Industrial Engineering** _____

Prof. Dr. Serhan Duran
Supervisor, **Industrial Engineering, METU** _____

Assoc. Prof. Dr. Ertan Yakıcı
Co-supervisor, **Industrial Engineering, MSÜ** _____

Examining Committee Members:

Prof. Dr. Esra Karasakal
Industrial Engineering, METU _____

Prof. Dr. Serhan Duran
Industrial Engineering, METU _____

Assoc. Prof. Dr. Mustafa Alp Ertem
Industrial Engineering, Çankaya University _____

Assoc. Prof. Dr. İsmail Serdar Bakal
Industrial Engineering, METU _____

Assist. Prof. Dr. Serhat Gül
Industrial Engineering, TEDU _____

Date: 06.12.2019

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Halenur Şahin Mahmutoğulları

Signature :

ABSTRACT

UTADIS BASED MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS FOR MEDICAL DIAGNOSIS PROBLEMS

Mahmutoğulları, Halenur Şahin
Ph.D., Department of Industrial Engineering
Supervisor: Prof. Dr. Serhan Duran
Co-Supervisor: Assoc. Prof. Dr. Ertan Yakıcı

December 2019, 268 pages

We develop hybrid methods that integrate multi-criteria decision making, evolutionary algorithms and machine learning to be used in medical diagnosis problems. The proposed models classify patients into two categories according to their disease status with the aim of obtaining high classification performances both classes under consideration.

First, we develop a Mixed-Integer Linear Programming approach, Parametrized Classification Model (PCM), which is based on UTADIS. By solving PCM multiple times with various values of a specific parameter, we obtain a set of solutions spread over the Pareto-optimal front in the space of true positive and true negative responses. Then, to combine strong aspects of these solutions, we integrate PCM with evolutionary algorithms, NSGA-II and RECGA, to tune the classification parameters acquired by PCM. NSGA-II favors non-dominated solutions in terms of sensitivity and specificity and RECGA aims to perform well particularly in situations where the incidence of the disease may be relatively low, such as general screening. We call the developed integrated models as PCM+NSGA-II and PCM+RECGA, respectively.

In order to observe the model performances, we try them with three different datasets which are about coronary stent patients and breast cancer. Furthermore, we apply several well-known machine learning algorithms to these datasets and compare the results with the results of PCM+NSGA-II and PCM+RECGA. Additionally, for the coronary stent dataset, the model performances are compared with those of cardiologists.

The results indicate that PCM+NSGA-II and PCM+RECGA are promising classification algorithms that can be used in medical decision support tools by medical experts.

Keywords: multi-criteria decision making, evolutionary algorithms, machine learning, medical diagnosis, rare event classification

ÖZ

TIBBİ TEŞHİS PROBLEMLERİ İÇİN UTADIS TEMELLİ ÇOK AMAÇLI EVRİMSEL ALGORİTMALAR

Mahmutoğulları, Halenur Şahin
Doktora, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Serhan Duran
Ortak Tez Yöneticisi: Doç. Dr. Ertan Yakıcı

Aralık 2019 , 268 sayfa

Bu çalışmada, tıbbi tanı problemleri alanında kullanılmak üzere, çok kriterli karar verme, evrimsel algoritmalar ve makine öğrenmesi yöntemlerini birleştiren hibrit yöntemler geliştiriyoruz. Önerilen modeller, incelenen her iki sınıfta da yüksek sınıflandırma performansları elde etmeyi amaçlayarak, hastaları durumlarına göre iki kategoride sınıflandırıyor.

İlk olarak, PCM olarak adlandırdığımız, UTADIS temelli bir karma tamsayılı programlama modeli geliştiriyoruz. PCM'i spesifik bir parametrenin çeşitli değerleri için birçok kez çözerek, doğru pozitif ve doğru negatif yanıtların alanında Pareto-optimal cepheye yayılmış bir dizi çözüm elde ediyoruz. Bu çözümlerin güçlü yönlerini birleştirmek için PCM ve evrimsel algoritmaları beraber kullanıyoruz. Bu amaçla, PCM'den elde edilen sınıflandırma parametrelerinin değerlerini, NSGA-II ve RECGA adlı evrimsel algoritmalar kullanarak ayarlıyoruz. NSGA-II, doğru pozitif ve doğru negatif sınıflandırma performansları açısından bir çözümün Pareto-optimalitesini yansıtan, baskılanamayan çözümleri öncelemektedir. RECGA ise, genel tarama gibi hastalığın görülme oranının göreceli olarak düşük olabileceği durumlarda özellikle iyi perfor-

mans göstermesini hedeflediğimiz bir başka evrimsel algoritmadır. PCM ile evrimsel algoritmaların entegrasyonu sonucu elde ettiğimiz modelleri, sırasıyla, PCM+NSGA-II ve PCM+RECGA olarak adlandırıyoruz.

Önerdiğimiz modellerin deneysel analizini üç farklı veri seti üzerinde yapıyoruz. Bu veri setlerinden ilki koroner stent implantasyonu yapılmış hastalar ile ilgili iken diğer iki veri seti ise meme kanseri ile ilgilidir. Buna ek olarak, bu veri setlerine bazı makine öğrenmesi yöntemlerini uyguluyoruz ve performanslarını önerilen modellerin performansları ile kıyaslıyoruz. Ayrıca, koroner stent veri seti için model performanslarını kardiyologların performansı ile karşılaştırıyoruz.

Elde ettiğimiz sonuçları incelediğimizde, PCM+NSGA-II ve PCM+RECGA'nın tıbbi karar destek aracı olarak kullanılabilecek, güvenilir ve etkin sınıflandırma yöntemleri olduğunu gözlemliyoruz.

Anahtar Kelimeler: çok kriterli karar verme, evrimsel algoritmalar, makine öğrenmesi, tıbbi teşhis, nadir olay sınıflandırma

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisors Prof. Serhan Duran and Assoc. Prof. Ertan Yakıcı for their guidance and support throughout my study.

I would also thank to Prof. Pınar Keskinocak and Assoc. Prof. Kamran Paynabar for their guidance, support and inspiration during my visit at Georgia Tech.

I am grateful to Prof. Esra Karasakal and Assoc. Prof. Alp Ertem for reading each part of this thesis and providing precious suggestions during my study. I also want to thank to Assoc. Prof. İsmail Serdar Bakal and Asst. Prof. Serhat Gül for accepting to be a member of my examination committee and for their valuable suggestions.

I am grateful to so many people for their unlimited support to me. I would like to thank Başak Yazar, Nihal Berktaş, Kamyar Kargar, Halil İbrahim Bayrak, Cemal İlhan, Cansu Gülcan, Haşim Özlü, Nur Timurlenk and Özlem Mahmutoğulları for being great friends. I would like to express my deepest gratitude to Özge Özgenç, Pınar Okutgen, Emine Topaloğlu and Ezgi İrem Bektaş for all valuable moments while sharing a childhood and a youth together.

I have also spent great time with my dear friends Fırat Kılcı, Işıl Koyuncu, Ömer Kerem Bekteş and Sweetie Koyuncu-Kılcı during my visit to Atlanta.

I thank to 15 anonymous doctors for performing ISR predictions.

During my study, I have been financially supported by TÜBİTAK programs 2211 and 2214. I am grateful to TÜBİTAK for providing this opportunity to me.

I would like to thank my dad Mahmut not just being a great dad but also a great colleague, my mom Dilek and my brother Alperen. I could not succeed without their endless love and support.

Finally, I am grateful to İrfan Mahmutoğulları for his endless love, support, encouragement and contribution to every single moment of my doctoral study.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

AHP	Analytical Hierarchy Process
A-SFM	Averaging Support Feature Machine
AIRS	Artificial Immune Recognition System
ANN	Artificial Neural Network
BMS	Bare Metal Stent
CABG	Coronary Artery Bypass Grafting
CART	Classification and Regression Trees
CNN	Combined Neural Network
CV	Cross validation
DBN	Deep Belief Network
DBN-NN	Deep Belief Network Path
DES	Drug Eluting Stent
DT	Decision Tree
ELECTRE IV	Elimination Et Choix Traduisant la Realité - Elimination and Choice Expressing Reality
FN	False Negative
FNR	False Negative Ratio
FP	False Positive
FPR	False Positive Ratio
GA	Genetic Algorithm
GRU-SVM	Gated Recurrent Unit Support Vector Machine
IBK	Instance Based K-Nearest Neighbor
K-NN	K-Nearest Neighbor
LASSO	Least Absolute Shrinkage and Selection

LDA	Linear Discriminant Analysis
LR	Logistic Regression
LSA	Logarithmic Simulated Annealing
LVQ	Learning Vector Quantization
M.H.DIS	Multi Group Hierarchical Discrimination
MACBETH	Measuring Attractiveness by a Categorical Based Evaluation Technique
MCDA	Multi Criteria Decision Analysis
MI	Myocardial Infarction
MLPNN	Multilayer Perceptron Neural Network
MOEA	Multi Objective Evolutionary Algorithm
MOMGA	Multi Objective Messy Genetic Algorithm
MRI	Magnetic Resonance Imaging
MSM	Multi Surface Method
NB	Naïve Bayes
NEFCLASS	Neuro Fuzzy Classification
NN	Neural Network
NPV	Negative Predictive Value
NSGA-II	Non-dominated Sorting Genetic Algorithm II
PAES	Pareto Archived Evolution Strategy
PCM	Parametrized Classification Model
PDA	Preference Disaggregation Analysis
pen-LR	Penalized Logistic Regression
PNN	Probabilistic Neural Network
PPV	Positive Predictive Value
PROAFTN	Procédure d'Affectation Floue pour la Problématique du Tri Nominal - Fuzzy Assignment Procedure for Nominal Sorting
PSO	Particle Swarm Optimization

PTCA	Percutaneous Transluminal Coronary Angioplasty
RECGA	Rare Event Classifier Genetic Algorithm
RF	Random Forest
RIAC	Rule Induction Through Approximate Classification
RIW-BPNN	Randomly Initialized Weight Back-Propagation Neural Network
RNN	Recurrent Neural Network
ROC	Receiver Operating Curve
RS_SVM	Rough Set-based Support Vector Machine
RS-BPNN	Rough Set Indiscernibility Relation Method
LS-SVM	Least Square Support Vector Machine
SMO	Sequential Minimal Optimization
SMOTE	Synthetic Minority Over-sampling Technique
SPEA	Strength Pareto Evolutionary Algorithm
SPECT	Single Photon Emission Computed Tomography
SVG	Saphenous Vein Graft
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
UTADIS	Utilities Additives Discriminantes
WBCO	Wisconsin Breast Cancer Original Dataset
WBDO	Wisconsin Breast Cancer Diagnostic Dataset

CHAPTER 1

INTRODUCTION AND MOTIVATION

Predicting the existence or absence of a disease has a crucial importance in health-care. There are various medical methods of diagnosis such as biopsy, ultrasound, MRI etc. However, most of these methods are expensive and/or carry risks for the patients. Moreover, some cases may have no apparent symptoms or clinical findings. Instead of using expensive methods or medical tests, operations research techniques and machine learning methods can be employed in medical decision support tools. These applications may help doctors to make predictions without creating additional risk and cost for patients. Moreover, for the cases where a general screening test is conducted among a population, the number of people carrying the disease is expected to be rare. For such cases, identifying the existence of the disease is harder and has greater importance.

For this purpose, in this thesis, we develop hybrid methods which integrate multi-criteria decision analysis, evolutionary algorithms and machine learning to be used in medical diagnosis problems and also perform good when incidence of the disease is relatively low in the population.

In Chapter 2, we give a literature review which covers studies on machine learning, prediction models in health-care, multi-criteria decision analysis, rare event classification and role of evolutionary algorithms in machine learning and multi-objective decision analysis.

In Chapter 3, we propose predictive classification methods where the patients are classified into two sets according to their disease status. The objective is to obtain high classification performances on both classes under consideration.

First, we propose the Parametrized Classification Model (PCM). It is a Mixed-Integer Programming (MIP) model and a variant of multi-criteria decision analysis method, UTADIS (UTilités Additives DIScriminantes)[3]. The objective of the model is to assign the best possible values to the decision variables with respect to the classifications given in the training set, in order to develop a set of additive utility functions. These additive utility functions are then used to classify patients into the predefined classes.

Unlike a classical UTADIS model, PCM has a parametric nature and it aims to minimize the number of false negative classification while keeping the number of false positive classification under a specified level of a parameter. Since, there is a trade-off between obtaining highest true positive and true negative responses from the model, changing this level and solving the linear programming model to optimality favors one objective while deteriorating the other. Thus, different levels for true positive and true negative responses are obtained each time. Therefore, the linear model creates a set of solutions spread over the Pareto-optimal front in the two dimensional space of true positive and true negative responses. In other words, PCM is used to obtain a set of solutions, some of which have high true positive and some have high true negative classification performances. Then, to combine the strong aspects of these solutions, we utilize evolutionary algorithms to tune the model parameters. By integrating multi-criteria decision analysis model, PCM, with evolutionary algorithms, we aim to develop models that have high classification performances in complex problems. To do so, first, the evolutionary algorithms derive new solutions through genetic operations, using the set of solutions obtained from PCM. Then, for each generation of solutions, they test the classification performances of the solutions with a validation set. Next, the evolutionary algorithms update the existing solutions by selecting the ones that can achieve good classification results in some aspects, where the goodness of a solution can be represented in many different ways (eg. high accuracy, high true positive rate, high positive predictive rate etc.). By this way, we aim to obtain novel combinations of model parameters such that their resulting classification have high true positive and true negative responses, simultaneously.

The first evolutionary algorithm developed to integrate with PCM is a multi-objective evolutionary algorithm, based on NSGA-II (Non-dominated Sorting Genetic Algo-

rithm II) [4]. It favors non-dominated solutions in terms of true positive and true negative classification performances. The purpose of the algorithm is to obtain solutions which can minimize false positive and false negative classification errors, simultaneously. The proposed algorithm is called as PCM+NSGA-II.

We also develop another solution method which is suitable for the problems with class imbalance. For such problems, when one class of observations is significantly less than the other, achieving high accuracy is possible by assigning all the observations to the class that constitutes the majority. In this case, a model would be totally inefficient to classify the rare observation. To overcome this drawback, we again consider using an evolutionary algorithm, called Rare Event Classifier Genetic Algorithm, (RECGA) together with the Mixed-Integer Linear Programming model PCM, and we call it as PCM+RECGA. It aims to achieve high true positive and true negative classification rates, even in the cases where one class of observations is significantly rare.

We apply PCM+NSGA-II and PCM+RECGA to three medical datasets and give the results of this experimental analysis. Additionally, to see the effect of integrating evolutionary algorithms with PCM, we compare the performances of PCM+NSGA-II and PCM+RECGA with the algorithms whose initial solutions are not obtained with PCM but random (Random+NSGA-II and Random+RECGA). Finally, we compare the performances of PCM+NSGA-II and PCM+RECGA with several machine learning algorithms.

In Chapter 4, we provide the results of the experimental analyses conducted in patient classification in terms of risk of restenosis after coronary stent implantation. In this context, we first determine the predictors by investigating the relevant literature and consulting with the experts. Then, we apply feature selection to find the most related set of coronary in-stent-restenosis predictors to build the simplest model and improve the prediction ability. We gather the data based on existing records of patients with coronary stents, from Ondokuz Mayıs University Hospital, Cardiology Department. We scan the records of 10,435 patients between the years 2005 and 2016 to find a set of patients who are eligible to be included in this study. The final dataset includes 303 observations. We test the performances of the models on this dataset, and also we compare them with the predictions of 15 cardiologists. We observe that the suggested

methods are effective and reliable decision support tools to classify patients in terms of in-stent-restenosis.

In Chapter 5, we report the results of the experimental analyses on two well-studied datasets about breast cancer (Wisconsin Breast Cancer Original Dataset [5] and Wisconsin Breast Cancer Diagnostic Dataset [6]). For both datasets, it is assumed that, the independent and dependent variables are the breast cancer predictors and the type of tumor (malignant or benign), respectively. By adjusting the rareness of malignant observations in the population from 35%-37% to 1%, we test the model performances in case of rare events. We observed that, our algorithms are promising classification algorithms and they are stronger alternatives when one class of observations is rare.

In Chapter 6, we propose concluding remarks as well as the possible future extensions of the existing study.

Part of the study reported in Chapters 3 and 4 is published in a relevant respected scientific journal [7], indexed by SCI.

CHAPTER 2

LITERATURE REVIEW

Since the proposed algorithms in this study are based on a multi-criteria decision analysis method, evolutionary algorithms and machine learning, our literature review focuses on these fields.

In the following subsections, first we review the machine learning literature in general and continue with prediction models in health-care, multi-criteria decision analysis literature, rare event classification models and the role of evolutionary algorithms in machine learning and multi-objective decision analysis, respectively.

2.1 Machine Learning

Machine learning has a vast area of application. Finance, manufacturing, medicine, medical diagnosis, telecommunication, chemistry, cognitive modeling, image recognition and speech recognition are some examples of these areas [8, 9].

A machine learning model is about experience, task and performance measure. Author Tom Mitchell states that "a computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at task in T, as measured by P, improved with experience E"[10].

Machine learning algorithms are categorized into groups of supervised, unsupervised and reinforcement learning methods. In the supervised learning, both input and output values are given where the latter are provided by a supervisor. The aim of the supervised learning algorithm is learning a mapping from the input to the output. In unsupervised algorithms, we only have input data and the aim is to discover some pat-

terns in it. In the reinforcement learning, the objective of the algorithm is to generate a policy, which is comprised of sequences of actions, by learning from past good action sequences. Some examples to reinforcement learning algorithms are game playing and robot navigating [8].

The models proposed in this study can be considered to be in the class of supervised learning algorithms. In the supervised learning, the given data set contains information about how a correct output should look like [10]. Therefore, the algorithm utilizes a training set to fit the parameters of the model. After the model parameters are obtained, a test set is used to measure the performance of the model. In some cases, a validation set is also used in order to tune the parameters [11].

Depending on the type of the output, supervised learning algorithms are classified as regression and classification. In regression problems, the algorithm provides continuous outcomes, whereas the outcomes are discrete in classification problems. The main supervised classification techniques are logic-based algorithms [12], perceptron-based techniques [12], statistical learning algorithms [12], kernel-based algorithms [12, 8], linear discrimination [8], and instance-based (non-parametric) learning [12].

In this study, the input is represented by a training set, which makes the proposed algorithms instance-based. They are used to classify the observations in two classes, thus they can be categorized also as classification algorithms.

2.2 Prediction Models In Health-care

In this section, we review the prediction models used in the field of health-care with a special emphasis on disease diagnosis. We find that the studies mainly focus on data driven, intelligent methodologies.

Statistical methods were commonly used on disease diagnosis in previous studies. However, due to their limitations on nonlinear and dependent data, data mining techniques and artificial intelligence become prominent [13, 14]. Data mining is defined as “the extraction of implicit, previously unknown, and potentially useful information from data [15]” and the process of discovering attractive patterns among the data

which makes sense in the decision making [16]. Data mining is utilized in some studies related with cardiac SPECT (single photon emission computed tomography) diagnosis, quality assessment of hemodialysis services and survival time prediction for kidney dialysis patients [16]. Machine learning is referred as the provider of the technical basis of data mining and it is used to extract information from the raw data [15]. It is also stated that, besides association, clustering, sequential patterns and similar time sequences, data mining can also be practiced through classification and prediction [17]. For classification purposes, statistics, decision trees, fuzzy sets, rough sets, neural networks and linear programming are widely used techniques [18, 19].

Tom Mitchell states that "although machine learning algorithms are central to the data mining process, it is important to note that the data mining process also includes other important steps such as building and maintaining the database, data formatting and cleansing, data visualization and summarization, the use of human expert knowledge to formulate the inputs to the learning algorithm and to evaluate the empirical regularities it discovers, and the eventual deployment of results. Thus data mining bridges many technical areas including databases, human-computer interaction, statistical analysis and machine learning algorithms" [20].

Appropriate computer-based information and decision support systems can help making clinical medical diagnosis at a reduced cost and they are valuable aids in achieving accurate results in medical diagnosis [21, 22]. The aim of these systems is enhancing rather than replacing the medical diagnosis decision of the physician [22].

One of the suggested approaches for this purpose is the K-nearest neighbors, which is a non-parametric pattern recognition method [16, 22]. The K-nearest neighbor approach is used in the diagnosis of lower back disorders, 30-day mortality and survival following acute myocardial infarction, and separating cancerous and non-cancerous breast cancer tumor masses [22].

Other pattern recognition methods that are used in the medical diagnosis are discriminant analysis [16] and Bayesian classifiers [16, 14], where naïve Bayes classifier is mentioned as a linear classifier in [23, 21, 24].

Meta-heuristic algorithms are also considered in medical diagnosis. Genetic algo-

rithm is referred as a data mining technique in some studies [16, 13, 25]. Neshat et al. propose a combination of particle swarm optimization and case-base reasoning for diagnosis of hepatitis disease [25].

Huang et al., Lin and Uzoka et al. consider case-base reasoning, which is defined as using old experiences to suggest solutions for the new cases [16, 13, 14]. These studies are about diagnosis and analysis of dysmorphic syndromes, assistance in making diagnosis and selection of a course of therapy. In their study, Huang et al. introduce a model which is developed for diagnosis and prognosis of chronic diseases. The proposed model integrates data mining and case-base reasoning [16].

Decision trees are referred as non-parametric supervised learning methods used for classification. They are also defined as inductive learning of symbolic rules, data mining tools and multi-stage decision making approaches [16, 21, 25]. A decision tree classifies the observations in the training set by construction and pruning operations. By this way, a decision tree classifier is able to find meaningful relations between the class labels and the set of observations which are used for training. Then, the acquired information is used to classify subsequent observations [26, 27]. Some of the areas where decision trees are used in medical literature are diagnosis of breast tumor in medical ultrasonic images [28], heart disease prediction [29] and diagnosis of type-II diabetes [30]. A study suggests a model for the liver disease diagnosis [13]. The proposed model is comprised of classification and regression tree (CART) and case-base reasoning. By the CART model, it is determined whether the patient suffers from the liver disease and by the case-base reasoning, the type of the liver disease is identified.

A collection of decision trees is called as random forest, which is an ensemble approach to build predictive models. Mangiameli et al. indicate that the predictive success of ensemble models is more accurate than single models [22]. The reason behind this idea is based upon the instable nature of single models due to changes in the learning set. The authors claim that ensemble of models are more robust. By combining a series of decision trees, random forest aims to increase prediction accuracy [31]. Lee et al. deal with the use of random forest for a lung nodule classification problem [32].

Another study that introduces ensemble strategies for medical diagnosis focuses on early detection and diagnosis of breast cancer. The authors analyze selection strategies of classification models to form ensembles and they compare the performances of a single model and ensemble of models [33].

Another method that can be used as a medical diagnosis aid is decision rules, which is an inductive learning of symbolic rules [16]. Moreover, Uzoka et al. [14] and Malmir et al. [34] consider rule-based programming and fuzzy rule-based decision support system, respectively.

It is claimed that, among the computer intelligence based methods used in medical diagnosis, neural networks are the most widely used [25]. Artificial neural network (ANN) is a non-parametric data mining technique and it is addressed in [16, 13, 22, 23] and [25] as a medical diagnosis tool. ANN has powerful pattern classification and pattern recognition capabilities, developed with the inspiration from the human brain neurological system as a data-driven self adaptive method and referred as a multivariate, non-linear, non-parametric statistical method [35]. Breast cancer, acute myocardial infarction, colorectal cancer, lower back disorders, drug/plasma concentration levels, hepatic cancer, sepsis, cytomegalovirus retinopathy, ovarian cancer, acute pulmonary embolism, micro calcification classification in digital mammograms and control of blood transfusion costs for surgery are some of the areas that neural networks are utilized [22]. Wu et al. suggest a three layer feedforward neural network with a back-propagation algorithm as a decision making tool for the analysis of mammographic data [36]. Malmir et al. discuss an adaptive neural-fuzzy inference system [34].

Another data mining technique, fuzzy sets, is addressed in [16] and [13]. Malmir et al. [34] propose online diagnostic application that uses fuzzy expert systems, fuzzy C-mean clustering method along with pattern recognition and adaptive neuro-fuzzy inference system along with the ANNs. The experiments of the models are conducted for diagnosis of kidney stone and kidney infection.

Huang et al. [16] and Lin [13] refer the inductive logic programming as another data mining technique and inductive learning of symbolic rules. Carrault et al. [37] address the use of inductive logic programming in arrhythmia recognition from elec-

trocardiograms.

Conforti and Guido propose a kernel-based support vector machine for medical diagnostic decision making problems. They propose an optimization based approach to learn the kernel function of support vector machine that performs best. They test the performance of their suggested model on breast cancer, heart disease, thyroid, ovarian cancer, leukemia and colon tumor datasets [38].

Some other studies consider rough sets [13], hypertext based systems, knowledge based technology, discriminant analysis and utility theory in the field of medical diagnosis [14].

Mangiameli et al. propose logistic regression to predict or diagnose acute myocardial infarction, coronary artery disease, liver metastases, gallstones, ulcers, mortality risk for reactive airway disease and breast cancer [22]. Logistic regression is a mathematical modeling approach which is used to identify relationship of several independent variables to a dichotomous dependent variable. In this way, it accomplishes predictive analysis [39]. A variant of it, the penalized logistic regression, is a combination of the logistic regression with a penalization of the L_2 norm of the coefficients. Due to quadratic penalization, it is expected to achieve a more robust fit when there is collinearity among variables, levels of discrete factors are sparse or high-order interaction terms exist [40]. It is also a promising tool when class imbalance is present (see [41, 42]).

Other medical diagnosis methods are Fisher linear discriminant analysis and kernel density. Fisher linear discriminant analysis is employed in diagnosis of coronary artery disease, acute myocardial infarction and breast cancer. Kernel density is utilized to differentiate malignant and benign cells taken from fine needle aspirates of breast tumors [22] .

Mangasarian et al. [43] proposed a breast cancer diagnosis tool based on image processing. The proposed classification procedure is a linear programming approach which separates malignant and benign samples. It is named as MSM-Tree since it is a variant of multi-surface method (MSM).

Zhang et al. introduce a rough set-based multi-criteria linear programming approach

for medical diagnosis. They are motivated by the shortcoming of multi-criteria linear programming model in reducing the dimension of input information space. Therefore, they integrate the rough set approach to the multi-criteria linear programming model to discover the hidden patterns among data and to eliminate the redundant dimensions of the information. The proposed approach is employed in diagnosis of breast cancer, heart disease and lung cancer [19].

The Analytic Hierarchy Process (AHP) is introduced by Saaty as a multi-criteria decision making approach, where the factors are arranged in a hierarchical manner [44]. Liberatore and Nydick deal with the usage of AHP in the field of medical diagnosis. They address its application to the sequential selection of diagnostic tests for the analysis of upper abdominal path and determining the overuse of endoscopy for low risk patients with acute upper gastrointestinal bleeding [45]. Uzoka et al. focus on the effectiveness of fuzzy and the AHP methods in diagnosis of malaria [14].

Pinheiro et al. and Brasil Filho et al. provide multi-criteria models as an aid to diagnose the Alzheimer's disease [46, 47]. Pinheiro et al. use the MACHBETH multi-criteria decision analysis method [46] and Brasil Filho et al. utilize two multi-criteria decision analysis classification approaches, PROAFTN and ELECTRE IV [47]. In the latter study, a genetic algorithm is applied for parameter optimization and its authors indicate that, in the multi-criteria decision analysis field, genetic algorithms are primarily used to control parameter optimization.

Feature extraction and hidden Markov models are other tools that are utilized to improve diagnostic systems [23], whereas discretization method and genetic search are available to eliminate redundant factors [24].

2.3 Multi-Criteria Decision Analysis

Our problem can also be investigated under multi-criteria classification/sorting problems. A classification/sorting problem aims to assign a set of alternatives into predefined groups. In the case of sorting, there is a preference relation among classes.

The main application areas of classification/sorting problems are medicine, pattern

recognition, human resources management, production systems management and technical diagnosis, marketing, environmental and energy management, ecology, financial management and economics. A detailed discussion on these application areas can be found in [48].

Criteria aggregation models and model development techniques are the two main issues that should be considered in the construction of a classification/sorting model. Outranking relation and utility function are referred as the main criteria aggregation models.

$U_g = \sum_{j=1}^{m} u_j(g_j), u_j(g_j) \in [0, 1]$, represents the simplest form of additive utility function where $u_j(g_j)$ is the marginal utility function of criterion g_j representing the "worth" of corresponding criterion in terms of utility term. In order to measure the performance of an alternative when all criteria are considered, the global utility $U(a_i)$ of an alternative a_i is calculated. Global utility values of alternatives are the measures which are used in classification/sorting of the alternatives into predefined groups. There are also utility thresholds representing a lower bound for belonging a specific class. The classification of an alternative is done by comparing the global utilities of an alternative with the utility threshold of each class under consideration [48]. This approach is called as UTADIS [49].

The other issue, model development technique, has two alternatives: direct and indirect model estimations. Model development includes specifications of weights of the evaluation criteria, preference-indifferences and veto thresholds. The preferences among the alternatives can be specified directly by the decision maker. These techniques are called as direct procedures. Whereas, in indirect procedures, the aim is to find preferential parameters which are as consistent as the decision maker's preferences with respect to previous decisions. In a similar manner, a training sample, which consists of previous decisions can be used. This approach is called as preference disaggregation analysis (PDA) and it uses regression-based techniques [50, 48]. In their study, Zopounidis and Doumpos stated that, within the multi-criteria decision analysis (MCDA) context, mathematical programming is a way to determine the optimal model parameters. The optimality measures for these mathematical models could be classification/sorting error rate of the alternatives and magnitude of viola-

tion/satisfaction of classification/sorting rules [48].

UTADIS is a supervised machine learning algorithm, which is also used in this thesis. More specifically, it is one of the instance based (non-parametric) MCDA approach developed to solve classification problems [3].

In the literature, variants of UTADIS are proposed for several problems such as, minimization of classification errors, maximizing the distances of correctly classified alternatives from the thresholds between classes, minimizing the number of misclassified alternatives. A variation of UTADIS is Multi-group Hierarchical Discrimination (M.H.DIS) method. M.H.DIS adapts the UTADIS model for more complex problems that have multiple groups and utilizes three mathematical models consecutively. The first model minimizes the magnitude of classification errors, the second model minimizes the number of misclassifications and the last one sharpens the acquired classification [49].

M.H.DIS is developed for sorting purposes and unlike the general form of a UTADIS model, instead of a single additive utility function for an alternative, there are $2(q-1)$ additive utility functions, in the existence of q classes[48]. M.H.DIS method determines the class of alternatives in a hierarchical manner by calculating utilities of the decision to classify an alternative into a specific class and a class lower than it. Then, by comparing these utility pairs, M.H.DIS determines the class which the alternatives under consideration must belong to. The procedure proceeds $q-1$ times until all groups are considered [51]. Note that, since the decision is based upon the comparison of utility pairs, unlike a classical UTADIS model, there is no threshold in M.H.DIS.

In a UTADIS model, which uses additive utility functions as the criteria aggregation method and PDA as the model development technique, the optimal values of the decision variables must be consistent with respect to the decision maker's preferences which are expressed in terms of previous decisions or a training sample [48]. Once the optimal values of the decision variables are found, they could be used to classify/sort new alternatives. Linear interpolation can be used to calculate the marginal utilities of new alternatives in test set by utilizing marginal utilities of alternatives in the training set [51].

2.4 Rare Event Classification

The success of a classification algorithm can be evaluated by the number of correctly classified alternatives, number of misclassified alternatives, classification errors and overall prediction accuracy. If there are only two classes (positive and negative) under consideration, the possible results of a classification should be one of these four cases: true positive, true negative, false positive, false negative. Then, the performance of a classification model under this setting can be measured with the true positive rate (sensitivity), true negative rate (specificity) as well as overall prediction ability (accuracy). However, in case of rare events, if the success of a prediction model depends on just the accuracy without consideration of sensitivity and specificity together, it is possible to obtain high accuracy rates by making correct predictions for the class of which members are frequently encountered in the population. Therefore, this situation may cause low prediction accuracy for the observations that belong to the class of rarely observed members. There are several studies in the literature which aim to overcome this problem caused by the class imbalance. Among them, many studies employ weighted support vector machine algorithms.

Support vector machine (SVM) is developed by Cortes and Vapnik, for two-group classification problems. The main objective is finding a linear hyperplane that distinctly classifies given data points [52]. Huang and Du propose a weighted SVM with different penalties of misclassifications for each class in the training sample. The authors claim that, "the equal penalty of misclassification for each training sample is one of reasons why the uneven training class sizes will result in classification biases". To overcome this drawback, they define penalties such that the ratio of penalties for different classes are equal to the inverse ratio of the training class sizes. They conduct experiments on Wisconsin Breast Cancer Diagnostic dataset [6], where the number of total observations is 569 with 357 benign and 212 malignant samples. 200 benign and 20 malignant samples are used to train the proposed algorithm, i.e. malignant observations are rare compared to benign observations where rareness level is 9% (malignant:benign = 20:200 = 1:10), and the rest of the observations are used in the test set [1].

Du and Chen [2] extend the work of Huang and Du [1] and they propose another

weighted SVM, v-SVM, where the misclassification penalties are different for each class in training sample as in [1]. The authors conduct experiments on the Wisconsin Breast Cancer Diagnostic dataset [6], with the same training and test configuration of [1].

Liu et al. state that when the class sizes in training sets are uneven, with SVM, undesirably biased classification errors found for the class with fewer observations in training set. To overcome this bias problem, a weighted SVM with genetic algorithm based parameter selection is proposed. The genetic algorithm determines the parameters related to regularization and the kernel function. Proposed algorithm is based on the idea of assigning larger weight factor corresponding to the class with fewer observations. The experimental analysis is conducted with IRIS dataset of UCI Repository [53]. The class sizes are 1000, 2000 and 3000 for class 1, class 2 and class 3; and the classification accuracies are 89.46%, 92.34% and 96.57%, respectively [54].

Yang et al. also propose a weighted SVM where the weights of the algorithm are generated by a robust fuzzy clustering technique, namely kernel-based probabilistic c-means algorithm. The problem is addressed as the outlier sensitivity problem. The experiments are conducted with an artificial data set and a benchmark dataset called "Twonorm", from the IDA benchmark repository [55]. The proposed algorithm is compared with SVM in terms of test error, for different number of mislabeled data points. It is observed that, as the mislabeled data points increase, the rate of increase in test error of proposed algorithm is slower than that of SVM [56].

There are methods in the literature used in the case of class imbalance other than weighted SVM. Li et al. propose particle swarm optimization, bat algorithm, and adaptive swarm balancing algorithm for imbalanced datasets. To reduce the imbalance in data, they introduce an algorithm called as SMOTE (Synthetic Minority Over-Sampling Technique). Experiments are conducted on ten datasets from UCI machine learning repository [57]. The imbalance ratios between majority class and minority class of these datasets range from 2.05:1 to 955.62:1. Rather than handling the imbalance in the original data set, they mainly focus on removing the imbalance in data to obtain high prediction accuracies for both classes [58] .

Wankhade et al. propose a hybrid of classification and clustering-based method for

problems with imbalanced classes. They use k-means, boosting and divide and merge methods. Suggested approaches are tested on KDDCup'99 [59], Car Evaluation [60], Cardiac Arrhythmia [61], Yeast [62], Adult [63], Shuttle [64] and Abalone [65] datasets from UCI machine learning repository. The proposed algorithm is able to deal with problems that have more than two classes. The majority class contains about more than 90% of the samples compared to the minority class. Average detection rate of the algorithm is 95% and its average false alarm rate is 0.4% [66].

2.5 Role of Evolutionary Algorithms in Machine Learning and Multi-Objective Decision Analysis

There is a wide application area of evolutionary algorithms, such as combinatorial optimization, expert systems, engineering applications, wired and wireless communication systems, medicine [67], design optimization, machine learning and parameter estimation problems [68]. Zhang et al. refer evolutionary computation as an optimization methodology inspired by the evolutionary mechanisms in the nature. The authors claim that, in the literature, while there are studies that consider evolutionary computing algorithms as a form of machine learning techniques, there are also studies that use machine learning techniques to enhance evolutionary computing algorithms [69].

Genetic algorithms, a subgroup of evolutionary algorithms, are mostly used to improve the prediction performances of machine learning methodologies either by feature elimination or parameter estimation. On the other hand, there are also many works that consider genetic algorithms as machine learning techniques rather than as an approach to enhance them.

As a parameter optimization tool, genetic algorithms can be employed by encoding a set of parameter values as a form of chromosome. The system optimizes the set of parameters which are represented by these chromosomes [70]. Goletsis et al. and Guvenir et al. use genetic algorithms in multi-criteria classification [68, 70]. Goletsis et al. discuss a multi-criteria sorting method to classify the cardiac beats as ischemic or not. In this work, genetic algorithm is applied in the training phase to determine

the model parameters, namely the thresholds and weight values [68]. In another study that uses genetic algorithm for parameter estimation, Guvenir et al. employ a multi-criteria inventory model and they determine the weights of criteria by a genetic algorithm [70]. De Jong also discusses the use of genetic algorithms as a parameter optimization/estimation tool [71]. The study of Kim et al. and the references therein propose genetic algorithm as a parameter estimation tool for ANN approach. Determining parameters of back-propagation network and training the weights of neural network are introduced as the genetic algorithm's two potential areas of usage [72].

Some works in the literature employ genetic algorithms as a feature elimination tool. Huang and Wang propose a SVM algorithm and they discuss the utilization of genetic algorithm as a feature selection tool and parameter optimization methodology. The proposed algorithm optimizes kernel and SVM regularization parameters and designs the fitness function by considering classification accuracy, the number of selected features and the feature cost such that a chromosome with high classification accuracy, a small number of features and low total feature cost reach a high fitness value [73]. Deekshatulu et al. propose a classification algorithm based on the K-nearest neighbor and genetic algorithm for heart disease. In this study, genetic algorithm is used as a feature elimination method and as a tool to rank the features which are used in classification [67].

Genetic algorithms are usually used to improve the performances of artificial intelligence techniques and to determine the architectural factors such as feature subset, number of hidden layers, activation functions and the connection weights between layers. For example, genetic algorithm can be used to feature discretization and determination of connection weights for ANN [74].

Chen refers evolutionary-based genetic algorithms as a machine learning and an artificial intelligence-based inductive learning technique [75]. Smith et al. propose an evolutionary algorithm as a classification method for the Parkinson's patients. The evolutionary algorithm employed in this study is referred as an implicit context representation of a Cartesian genetic programming [76].

Padgorolec and Kokol provide a self-adapting evolutionary algorithm for the induction of decision trees. The suggested model is used for diagnostic process optimiza-

tion. The model intends to minimize the number of examinations, select the most appropriate examination for a specific patient, optimize examination schedule and maximize the equipment reliability. Diversity of genetic preservation is considered as one of the most significant features of a successful evolutionary algorithm [77]. Another study that deals with the use of genetic algorithm for a decision tree induction algorithm is conducted by Turney [78]. In this study, each individual in the population represents one set of biases and genetic algorithm is used to evolve a population of biases for a decision tree induction algorithm. Corcoran and Sen propose a supervised classification problem. It is an optimization problem whose aim is to develop a rule set that maximizes the number of correct classifications of training set instances. The authors use genetic algorithm to evolve structures representing sets of classification rules [79].

In multi objective programming, genetic algorithms are used to identify non-dominated (Pareto-optimal) solutions [68]. In the existence of multiple objectives, the concepts of dominance and Pareto-optimality take the place of the conventional optimality theories. A solution x said to dominate solution y if and only if x is as good as y in terms of all objectives and better in at least one objective. Pareto-optimal set is the non-dominated subset of all feasible solutions [80]. Convergence to the Pareto-optimal front and maintaining a diverse set of solutions are the two main goals of multi-objective optimization algorithms. Evolutionary approaches suit well to the multi-objective problem characteristics [81]. Multi-objective evolutionary algorithms (MOEAs) work with a population of solutions and while preserving the diversity of solutions, they find multiple non-dominated solutions in a single run [4].

There are various types of MOEAs such as penalty based approaches, non-elitist and elitist MOEAs. Their application areas are also various. In their book, Coello et al. categorize these application areas as engineering, scientific, industrial and miscellaneous. Medicine is identified as a sub-category of scientific applications where classification and prediction is considered under miscellaneous applications [82]. Among MOEAs, elitist approaches have certain advantages. As a part of elitism, the solutions of current generation are compared with previously found best non-dominated solutions. By this way, it helps to obtain better convergence by ensuring the preservation of good solutions once they have been found. Additionally, elitism results in positive

contribution to the speed of genetic algorithm [4]. NSGA-II [4], strength Pareto evolutionary algorithm (SPEA)[83], Pareto archived evolution strategy (PAES) [84] and multi-objective messy genetic algorithm (MOMGA) [85] are the main elitist MOEAs.

NSGA-II is an elitist non-dominated sorting based multi-objective evolutionary algorithm in which solutions are ranked according to their non-domination and assigned to the fronts. The aim of the algorithm is to converge near the true Pareto-optimal set in the presence of multiple objectives in the problem. Together with non-domination, the algorithm also promotes diversity preservation using crowding comparison. If two solutions have different non-domination ranks (i.e. belongs to different non-dominated fronts), the solution with the lower (better) rank is given priority to be selected for the next generation. If two solutions have the same non-domination ranks (i.e. belong to the same front), the solution which is located in a less crowded region is favored in order to preserve diversity. Due to the elitist approach, the solutions of current generation are compared with previously found best non-dominated solutions. The algorithm aims to terminate with a set of solutions that converge to the true Pareto-optimal front [4].

After the literature review presented above, we can now state the position of our study in the literature. In this study, we develop methods for binary classification that integrates multi-criteria decision analysis and evolutionary algorithms. When the position of PCM utilized in this thesis is investigated in the context of MCDA in detail, it can be said that, type of the problem it addresses is classification/sorting and as the criteria aggregation model it uses additive utility functions. To specify the model parameters, PCM uses Mixed-Integer Linear Programming as the mathematical programming formulation. In order to measure the classification/sorting optimality with respect to the given classifications of alternatives, it evaluates the number of false positive and false negative classifications. In particular, while it minimizes the number of false negative classifications in the objective function, via a constraint, it forces to keep the number of false positive classifications under a certain level. However, as it is indicated by Conway et al. the linear models' classification performances are promising only if the alternatives are perfectly separable [86]. Thus, to develop classification algorithms which also perform good in the existence of more complex problems, we introduce a novel approach that integrates MCDA with evolutionary algorithms.

In this context, we develop PCM as a parametric model. In this way, PCM returns a set of solutions (instead of a single solution) spread over the Pareto-optimal front in the space of true positive and true negative responses. Thus, while some of these solutions have high sensitivity, some have high specificity. Then, the evolutionary algorithms are used to diversify the solutions and improve the classification performances.

Throughout this study, we look for answers to the following research questions:

- Does the integration of evolutionary algorithms with the Mixed-Integer Linear Programming model PCM provide better results than just randomly generating the initial solution set of evolutionary algorithms?
- Can the proposed classification algorithms compete with the well-known methodologies in the literature, in terms of prediction performance?
- What are the pros and cons of using the proposed classification methods? Are they promising for the cases where one class of observations is rare compared to other, such as general screening?

CHAPTER 3

MODEL DEVELOPMENT

We begin this chapter with definitions and descriptions of performance measures used to evaluate a classification algorithm. Then, we outline the prediction procedure of the classification algorithms developed in this study. Next, we explain the PCM in detail. After that, we discuss the basic characteristics of the proposed evolutionary algorithms and we explain PCM+NSGA-II and PCM+RECGA as classification algorithms, in a comprehensive manner. At the end of the chapter, the hyper-parameter optimization process conducted for the proposed model is explained, as well.

Throughout this chapter, the existence and absence of a disease is numerically represented by 1 and 0 as the values of the binary response variable for each of the patients (observations), whereas the words "positive" or "negative" define the same status, respectively. Possible results of classification are presented in Table 3.1.

Table 3.1: Possible Results of Classification (Contingency Matrix)

		Actual Class	
		1	0
Predicted Class	1	True Positive (TP)	False Positive (FP)
	0	False Negative (FN)	True Negative (TN)

We define sensitivity (true positive rate), specificity (true negative rate), accuracy, Positive Predictive Value (PPV), Negative Predictive Value (NPV), False Positive Ratio (FPR) and False Negative Ratio (FNR) which will be used to evaluate the success of the classification algorithms. These values are calculated as indicated in Equations 3.1 - 3.7 below, where TP (True Positive) and TN (True Negative) are the sets of pa-

tients that are classified accurately, and FP (False Positive) and FN (False Negative) are the sets of patients that are classified incorrectly.

$$Sensitivity = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}|}. \tag{3.1}$$

Sensitivity is defined as the proportion of true positives that are correctly identified by a model and it represents the success of the model in detecting the existence of a disease [87].

$$Specificity = \frac{|\text{TN}|}{|\text{TN}| + |\text{FP}|}. \tag{3.2}$$

Specificity is the proportion of the true negatives that are correctly identified by a classifier and it estimates the success of the model to identify negative observations [87].

A good classifier is the one that has both high sensitivity and specificity. As the classifier's ability to differentiate the classes correctly increases, its sensitivity and specificity increases, too. [88]. Sensitivity and specificity are inversely proportional [89].

$$Accuracy = \frac{|\text{TP + TN}|}{|\text{TP + TN + FP + FN}|}. \tag{3.3}$$

Accuracy represents the rate of correct classifications [87].

$$PPV = \frac{|\text{TP}|}{|\text{TP}| + |\text{FP}|}. \tag{3.4}$$

Positive predictive value of a classifier is the likelihood that an observation classified as positive actually has the disease [88], [89].

$$NPV = \frac{|\text{TN}|}{|\text{TN}| + |\text{FN}|}. \tag{3.5}$$

Negative predictive value of a classifier is the likelihood that an observation classified

as negative actually does not have the disease [88], [89].

$$FPR = 1 - Specificity. \tag{3.6}$$

$$FNR = 1 - Sensitivity. \tag{3.7}$$

To incorporate sensitivity and specificity values in a single measure, we define a combined performance measure, Fscore:

$$Fscore = \frac{2 \times Sensitivity \times Specificity}{Sensitivity + Specificity}$$

$$= \frac{2 \times |\text{TP}| \times |\text{TN}|}{2 \times |\text{TP}| \times |\text{TN}| + |\text{TP}| \times |\text{FP}| + |\text{TN}| \times |\text{FN}|}.$$

Fscore is the harmonic mean of sensitivity and specificity. Fscore $= 0$, whenever sensitivity or specificity is zero, and Fscore $= 1$, when sensitivity and specificity are one. Thus, it takes a value in the interval of $[0, 1]$. It takes lower values as the difference between sensitivity and specificity grows.

In the literature, another combined performance measure, Fmeasure is commonly used. It is defined as

$$Fmeasure = \frac{2 \times Precision \times Recall}{Precision + Recall},$$

where $Precision = PPV$ and $Recall = Sensitivity$, according to the given terminology. Thus,

$$Fmeasure = \frac{2 \times |\text{TP}|}{2 \times |\text{TP}| + |\text{FP}| + |\text{FN}|}.$$

However, there are some shortcomings of the Fmeasure. It does not take into account the true negatives. It focuses on one class only and it is biased by the class that constitutes majority [90].

Let us clarify these shortcomings with the following examples.

Example 1: Assume that, there are 100 positive and 100 negative observations in Case #1. In Case #2, the number of positive observations is equal to the previous case, but there are one million additional negative observations. Assume that, Table 3.2 presents the contingency matrices after classification of the observations for both cases:

Table 3.2: Contingency Matrices of Example 1

Case #1		Actual Class 1	Actual Class 0
Predicted Class	1	100	50
	0	0	50

Case #2		Actual Class 1	Actual Class 0
Predicted Class	1	100	50
	0	0	1000050

Some performance indicators for Case #1 are as follows:

$$Sensitivity = Recall = 1,$$
$$Specificity = 0.50,$$
$$Accuracy = 0.75,$$
$$Precision = 0.67,$$
$$Fmeasure = 0.80,$$
$$Fscore = 0.67.$$

However, the same performance indicators for Case #2 are:

$$Sensitivity = Recall = 1,$$
$$Specificity = 1,$$
$$Accuracy = 1,$$
$$Precision = 0.67,$$
$$Fmeasure = 0.80,$$
$$Fscore = 1.$$

Note that, in Case #2, even the model correctly classifies one million additional observations, Fmeasure does not change. This is because its formulation does not consider

true negatives. However, Fscore properly reflects the models' capability of classifying negative observations, for both cases.

Example 2: Assume there are 100 positive and 10 negative observations and a model classifies all of them as positive. Table 3.3 presents the contingency matrix.

Table 3.3: Contingency Matrix of Example 2

		Actual Class	
		1	0
Predicted Class	1	100	10
	0	0	0

According to the given classifications, the relevant performance indicators are as follows:

$$
\begin{aligned}
&Sensitivity = Recall = 1,\\
&Specificity = 0\\
&Accuracy = 0.91,\\
&Precision = 0.91,\\
&Fmeasure = 0.95,\\
&Fscore = 0.
\end{aligned}
$$

As it is observed, even when none of the negative observations are classified properly, Fmeasure is quite high. However, Fscore takes value of zero which indicates this absolute misclassification.

Example 3: Assume in Case #1 and Case #2, there are one million negative and 10 positive observations. In both cases, the models' performances on positives are perfect. In Case #1 and Case #2, two and 10 negative observations are classified incorrectly, respectively.

The contingency matrices for this example are given in Table 3.4.

Table 3.4: Contingency Matrices of Example 3

Case #1		Actual Class: 1	Actual Class: 0
Predicted Class	1	10	2
	0	0	999998

Case #2		Actual Class: 1	Actual Class: 0
Predicted Class	0	10	10
	0	0	999990

According to the given classifications, the performance indicators for Case #1 are as follows:

$$Sensitivity = Recall = 1,$$
$$Specificity = 1,$$
$$Accuracy = 1,$$
$$Precision = 0.83,$$
$$Fmeasure = 0.91,$$
$$Fscore = 1.$$

However, same performance indicators for Case #2 are:

$$Sensitivity = Recall = 1,$$
$$Specificity = 1,$$
$$Accuracy = 1,$$
$$Precision = 0.50,$$
$$Fmeasure = 0.67,$$
$$Fscore = 1.$$

Even though there is no significant amount of difference in the models' successes for these two cases, Fmeasure performances of the models are significantly different.

Example 4: There are one positive and 150 negative observations. Assume that, the

model classifies the only positive observation correctly, and it misclassifies only three of the 150 negative observations.

The contingency matrix of the given classification is presented in Table 3.5.

Table 3.5: Contingency Matrix of Example 4

		Actual Class	
		1	0
Predicted Class	1	1	3
	0	0	147

The performance indicators are:

$$Sensitivity = Recall = 1,$$
$$Specificity = 0.98,$$
$$Accuracy = 0.98,$$
$$Precision = 0.25,$$
$$Fmeasure = 0.40,$$
$$Fscore = 0.99.$$

Since precision is low, Fmeasure also takes a low value. The precision of 0.25 indicates that, the probability that a patient classified as positive by the model actually has the disease is 25%. Even this rate is quite low, there is no false negative observation. Thus, there is no risk in terms of human health but only financial burden occurs. This indicates that, for three patients who actually do not have the disease, since the model indicates them as positive they have to go under further investigation.

Our priority in this study is human health rather than the financial burden. Thus, Fscore is selected to be used in this thesis. Also, Fscore is more successful to evaluate the models' capability of distinguishing between classes, as seen clearly in Examples 1-4.

There are some other studies in the literature that evaluate the model performances using the harmonic mean of sensitivity and specificity (Fscore) [91], [92], [93], [94].

The measures which consider sensitivity and specificity together are generally used to evaluate performance of a model in separating positive and negative observations. In one of these studies, the author specifies that the harmonic mean of sensitivity and specificity is a ROC (receiver operating curve) based measure where it reflects the trade-off between true positive and false positive rate (i.e. sensitivity and 1-specificity) and it is desirable to deal with imbalanced data [92]. In another study, the authors refer harmonic mean of sensitivity and specificity as one of the operating point selection strategies to seek the point that maximizes the harmonic mean on the ROC curve [91].

In the following paragraphs, we give an overview of the algorithms proposed in this study. Recall that, PCM tries to fit its model parameters consistent as much as possible with the given classifications in the training set. By solving PCM, we obtain a set of solutions which consists of factor weights to be used in the utility functions to classify the patients. We run the model with different L values, where L stands for the number of false positive classifications that the model allows. Thus, for different values of L, the solutions have different sensitivity and specificity values. For small values of L, PCM allows less false positive classifications and thus the solutions have higher sensitivity. However, since there is a trade-off between sensitivity and specificity, specificity values of these solutions are relatively low. For the cases where L takes larger values, the model finds solutions with high specificity and low sensitivity. Since some of these solutions are characterized with their high sensitivity and the others with their high specificity, we utilize evolutionary algorithms NSGA-II and RECGA after PCM to achieve good solutions with both high sensitivity and high specificity, simultaneously. Thus, note that, PCM is not developed for prediction purposes, but only to generate the initial solution set which is then improved by the evolutionary algorithms.

Once the integrated algorithms, PCM+NSGA-II and PCM+RECGA provide a set of solutions, these solutions are used to classify a set of patients that the models have never seen before. For this purpose, the factor weights for the patients in test set are calculated by linear interpolation. Then, their additive utilities of having and not having the disease are found (U and $\widetilde{U}$, respectively). The additive utility values of a patient are associated with the values that he/she has in terms of each disease

predictor. Final class prediction of a patient p is made by comparing these additive utility pairs for each solution. If the number of solutions where $U(p) \geq \widetilde{U}(p)$ is greater than $U(p) \leq \widetilde{U}(p)$, the final prediction for the patient p becomes positive. In other words, the final class prediction of an observation is made by majority voting. Since the final decision is not based on a single solution, it can be said that, majority voting can contribute to have a more robust decision making process.

All of the notations used in the introduced models are listed in Table 3.6. The pseudo-code given in Algorithms 1 and 2 summarize the general prediction procedure of the classification algorithms developed in this study and the Figure 3.1 illustrates the procedure followed in Algorithm 1.

Table 3.6: Table of Notations for PCM, NSGA-II and RECGA

$\mathcal{F}$	set of factors, $\mathcal{F} = \{1, 2, ..., F\}$
$\mathcal{S}$	training set
$\mathcal{S}^+$	set of positive observations in training set
$\mathcal{S}^-$	set of negative observations in training set
$\mathcal{V}$	validation set
$\mathcal{V}^+$	set of positive observations in validation set
$\mathcal{V}^-$	set of negative observations in validation set
$\widetilde{\mathcal{S}}$	test set
O_f	set of all values of factor f that appear in $\mathcal{S}$,
	$O_f = \{o_{f_1}, o_{f_2}, ..., o_{f_{df}}\}$ s.t $o_{f_1} < o_{f_2} < ... < o_{f_{df}}$
	o_{f_i}: i^{th} value of factor f, when factor values are in ascending order
$x_f(p)$	value of factor f for observation p
$x(p)$	vector of factors of observation p
	$x(p) = (x_1(p), x_2(p)..., x_F(p))$
$u_f(.)$	utility function of factor f for positive classification
	$u_f(o_{f1}) = 0 \; \forall f \in \mathcal{F}$, $u_f(o_{f_i}) = u_f(o_{f_{i-1}}) + w_{f(i-1)} \; \forall i \in 2, ..., df$
$\widetilde{u}_f(.)$	utility function of factor f for negative classification
	$\widetilde{u}_f(o_{f_{df}}) = 0 \; \forall f \in \mathcal{F}$, $\widetilde{u}_f(o_{f_i}) = \widetilde{u}_f(o_{f_{i+1}}) + m_{f(i)} \; \forall i \in df - 1, ..., 1$
$U(p)$	utility function of observation p for positive classification,
	that is $U(p) = \sum_{f=1}^{F} u_f(.)$
$\widetilde{U}(p)$	utility function of observation p for negative classification,
	that is $\widetilde{U}(p) = \sum_{f=1}^{F} \widetilde{u}_f(.)$
w_{fi}	weight increase in utility u_f due to the i^{th} interval of factor f
	$w_{fi} \geq t$ and $w_{fi} = u_f(o_{f_{i+1}}) - u_f(o_{f_i})$
m_{fi}	weight increase in utility $\widetilde{u}_f()$ due to the i^{th} interval of factor f
	$m_{fi} \geq t$ and $m_{fi} = \widetilde{u}_f(o_{f_i}) - \widetilde{u}_f(o_{f_{i+1}})$
W	set of weight vectors: $W = \{w_f : f \in \{1, ..., F\}\}$
	where $w_f = (w_{f1}, w_{f2}, ..., w_{df-1})$
M	set of weight vectors: $M = \{m_f : f \in \{1, ..., F\}\}$
	where $m_f = (m_{f1}, m_{f2}, ..., m_{df-1})$

Table 3.6: Table of Notations for PCM, NSGA-II and RECGA

$Ind(p)$	false classification indicator for observation p
	$Ind(p) = 1$ if classification is false; 0 o.w.
$e(p)$	error term for observation p, $e(p) \in R^+$
s, t	small positive constants
L	false positive classification allowance
$PCM(L)$	PCM model with false positive classification allowance of L
(W$^*(L)$,M$^*(L)$)	Optimal solution of $PCM(L)$
$y(p)$	actual value for the response variable of observation p, $y(p) \in \{0, 1\}$
$\widetilde{y}(p)$	predicted value for the response variable of observation p, $\widetilde{y}(p) \in \{0, 1\}$
p_{rc}	probability of real crossover
p_{lc}	probability of linear crossover
p_m	probability of mutation
P_t	set of parent solutions at generation t
Q_t	set of offspring that derived from P_t, at generation t
R_t	combined set of parent and offspring population at generation t
Fr	set of all fronts
Fr_k	k^{th} front
PopulationSize	size of a generation of *NSGA-II*
GenerationSize	size of solutions selected to be carried to next generation
NumberOfGenerations	generation number of *NSGA-II* until termination
$threshold$	threshold for fitness function (Fscore) to be carried to next generation
minFinalSetSize	lower limit of number of final set of solutions of *RECGA*
PopulationSize	size of a generation of *RECGA*
$\mathcal{X}$	a set of solutions
$CP(p)$	counter for positive predictions
$CN(p)$	counter for negative predictions

Algorithm 1 General Prediction Procedure of a Classification Algorithm

for each $p \in \widetilde{\mathcal{S}}$ **do**
    Initialize the counter for positive and negative predictions $CP(p) = 0$ and $CN(p) = 0$, respectively.
end for
Solve *the classification algorithm*, and let $(W^i, M^i), i \in \{0, 1, ...n\}$ be the final set of solutions generated by *the classification algorithm*.
for each (W^i, M^i) **do**
    Calculate utility functions $u_f(\cdot)$ and $\widetilde{u}_f(\cdot)$ for each factor using the procedure *GenerateUtilityFunctions*
    for $p \in \widetilde{\mathcal{S}}$ **do**
        Set total utility value $U(p) = 0$ and total disutility value $\widetilde{U}(p) = 0$
        for each $f \in \mathcal{F}$ **do**
            Calculate utility functions $u_f(x_f(p))$ and $\widetilde{u}_f(x_f(p))$ for each factor using the procedure *GenerateUtilityFunctions*
            $U(p) \leftarrow U(p) + u_f(x_f(p))$
            $\widetilde{U}(p) \leftarrow \widetilde{U}(p) + \widetilde{u}_f(x_f(p))$
        end for
        if $U(p) \geq \widetilde{U}(p)$ **then**
            $CP(p) \leftarrow CP(p) + 1$
        else
            $CN(p) \leftarrow CN(p) + 1$
        end if
    end for
end for
for $p \in \widetilde{\mathcal{S}}$ **do**
    if $CP(p) \geq CN(p)$ **then**
        Prediction for observation p is positive, that is, $\widetilde{y}(p) = 1$
    else
        Prediction for observation p is negative, that is, $\widetilde{y}(p) = 0$
    end if
end for

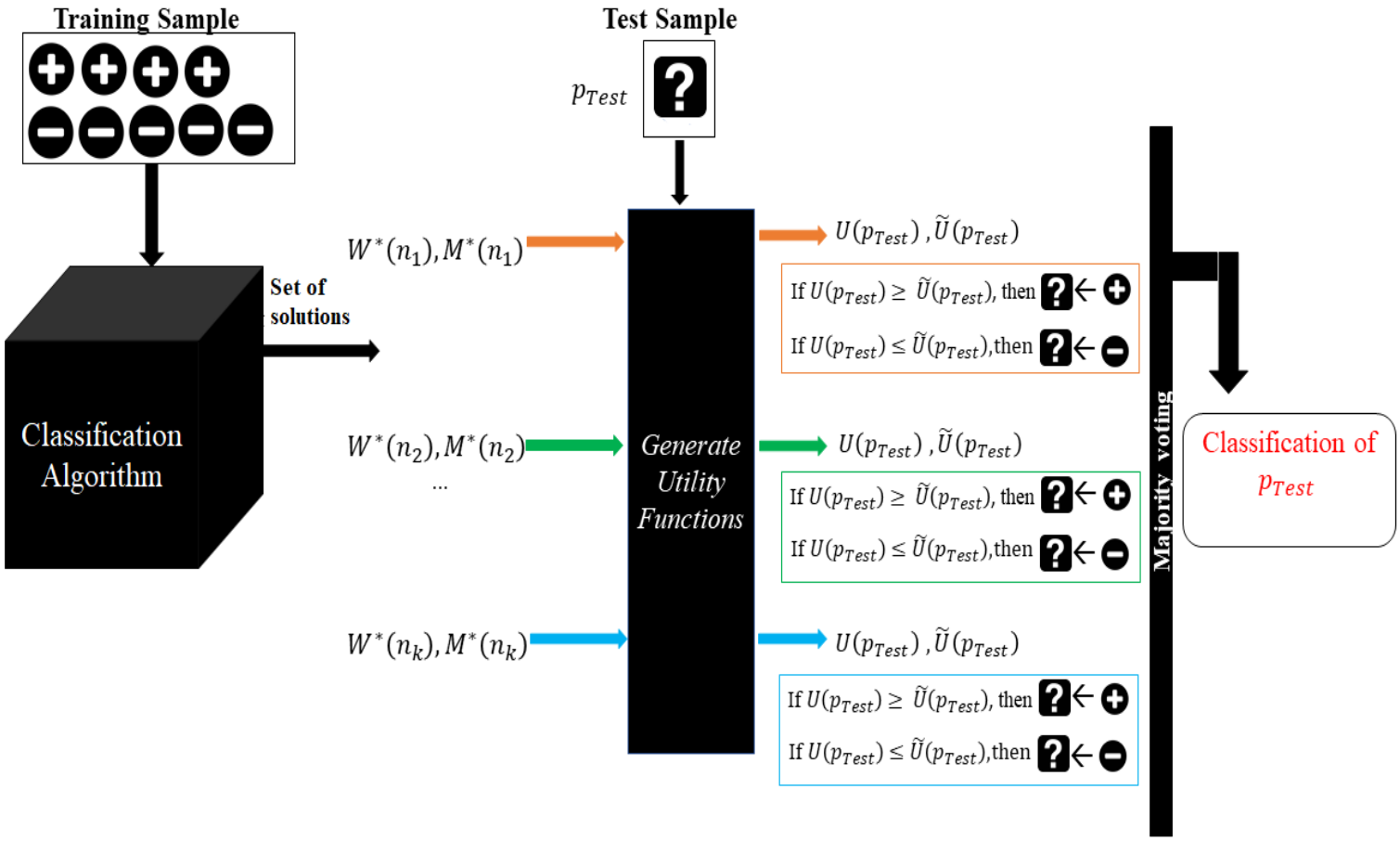


Figure 3.1: Prediction Procedure of a Classification Algorithm

Algorithm 2 *GenerateUtilityFunctions*

for each $f \in \mathcal{F}$ **do**
  $u_f(o_{f_1}) = 0$
  $\widetilde{u}_f(o_{f_{df}}) = 0$
  for each $i \in \{2, \ldots, df\}$ **do**
    $u_f(o_{f_i}) = u_f(o_{f_{i-1}}) + w_{f(i-1)}$
  end for
  for each $i \in \{df - 1, \ldots, 1\}$ **do**
    $\widetilde{u}_f(o_{f_i}) = u_f(o_{f_{i+1}}) + m_{fi}$
  end for
end for
for each $i \in \{1, 2, \ldots, df - 1\}$ **do**
  For any $a \in \mathbb{R}$, with $o_{f_i} < a < o_{f_{i+1}}$, calculate $\gamma \in (0, 1)$ that satisfies $a = \gamma o f_i + (1 - \gamma) o_{f_{i+1}}$
  $u_f(a) = \gamma u_f(o_{f_i}) + (1 - \gamma) u_f(o_{f_{i+1}})$
  $\widetilde{u}_f(a) = \gamma \widetilde{u}_f(o_{f_i}) + (1 - \gamma) \widetilde{u}_f(o_{f_{i+1}})$
end for
For any real number $a \notin [o_{f_1}, o_{f_{df}}]$, calculate $u_f(a)$ and $\widetilde{u}$ by linear regression using points $(o_{f_1}, u_f(o_{f_1})), (o_{f_2}, u_f(o_{f_2})), \ldots, (o_{f_{df}}, u_f(o_{f_{df}}))$ and $(o_{f_1}, \widetilde{u}_f(o_{f_1})), (o_{f_2}, \widetilde{u}_f(o_{f_2})), \ldots, (o_{f_{df}}, \widetilde{u}_f(o_{f_{df}}))$, respectively.

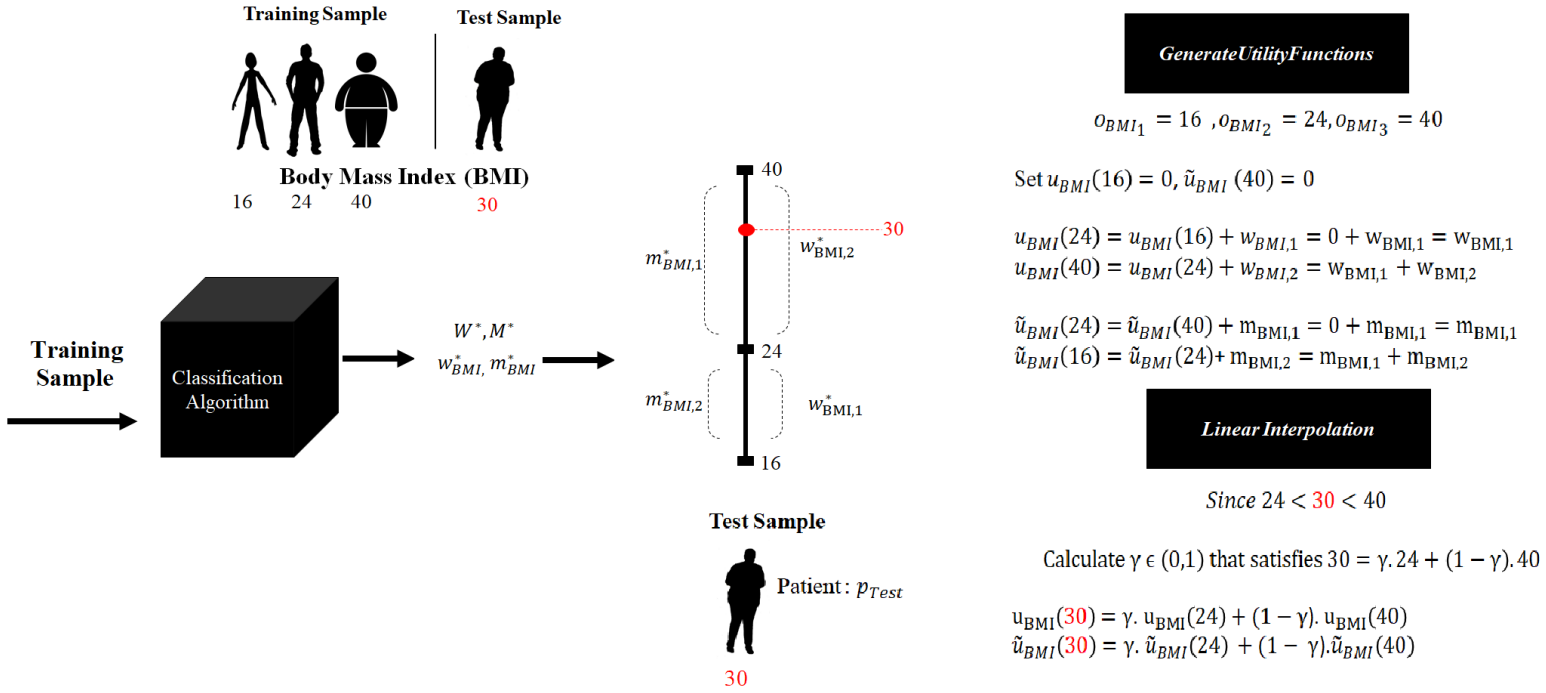


Figure 3.2: *GenerateUtilityFunctions* Algorithm

The Figure 3.2 illustrates *GenerateUtilityFunctions* algorithm for an example. Suppose the only factor to consider is body mass index (BMI) and there are three observations in the training set. Assume a classification algorithm gives a solution (W^*, M^*) due to the classifications of observations given in the training set. The resulting solution (W^*, M^*)=$\begin{bmatrix} w^*_{BMI,1} & w^*_{BMI,2} \end{bmatrix}$, $\begin{bmatrix} m^*_{BMI,1} & m^*_{BMI,2} \end{bmatrix}$ represents the incremental weights correspond to each factor value.

Due to the monotonicity assumption (which will be explained later) higher factor values are more likely to have the disease. Then, the thinnest patients' utility for positive classification and the fattest patients' utility for negative classification are zero (i.e. $u_{BMI}(16) = 0$, $\widetilde{u}_{BMI}(40) = 0$). Once the *GenerateUtilityFunctions* algorithm assigns these values, it calculates the utilities of each factor value for negative and positive classification, by summing up the incremental weights. Since the BMI of the patient in the test sample is different than the BMIs of patients in training sample, his corresponding utilities are calculated by linear interpolation that is part of the *GenerateUtilityFunctions* algorithm.

Now, let us explain the Parametrized Classification Model (PCM), which is solved to bring a set of solutions. Note that, a solution is represented with (W, M) and it is comprised of weights w_{fi} and m_{fi}, which are the decision variables of the PCM.

3.1 Parametrized Classification Model (PCM)

PCM is a UTADIS based MCDA classification model, where UTADIS is a method which aims to determine the thresholds of groups in a way to minimize the classification error. However, since PCM identifies the class of an alternative by comparing utility pairs, there is no threshold to be determined. According to this characteristic, the model can also be considered as a variant of M.H.DIS method. Recall that, M.H.DIS creates $2(q-1)$ additive utility functions to classify alternatives into q classes. In our problem, since there are two classes under consideration (existence of the disease and absence of the disease), two utility functions, U and $\widetilde{U}$, which represent the utility function that characterizes the category of patients with and without disease, respectively, are defined. Since the model determines the class of an alter-

native by comparing the corresponding global utility pairs in a hierarchical manner, the classification problem we are dealing is addressed as a sorting problem with two classes.

This approach aims to find preferential parameters consistent with the decision maker's preferences in terms of previous decisions, as much as possible. Thus, a training sample is used as a model of decision maker's preferences as in [50], [48].

Using the notations defined at Table 3.6, PCM is formulated as follows:

$$\text{min} \quad z_{PCM(L)} = \sum_{p \in \mathcal{S}^+} Ind[p] \tag{3.8}$$

$$\text{s.t.} \quad \sum_{f=1}^{\mathcal{F}} \sum_{i=1}^{j-1} w_{fi} - \sum_{f=1}^{\mathcal{F}} \sum_{i=j}^{df-1} m_{fi} + e(p) \geq s \quad \forall p \in \mathcal{S}^+, j : x_f(p) = o_{fj} \in O_f \tag{3.9}$$

$$\sum_{f=1}^{\mathcal{F}} \sum_{i=j}^{df-1} m_{fi} - \sum_{f=1}^{\mathcal{F}} \sum_{i=1}^{j-1} w_{fi} + e(p) \geq s \quad \forall p \in \mathcal{S}^-, j : x_f(p) = o_{fj} \in O_f \tag{3.10}$$

$$e(p) - \mathbf{M} \cdot Ind[p] \leq 0 \quad \forall p \in \mathcal{S}, \tag{3.11}$$

$$\sum_{p \in \mathcal{S}^-} Ind[p] \leq L \tag{3.12}$$

$$e(p) \geq 0 \tag{3.13}$$

$$w_{fi} \geq t, \quad m_{fi} \geq t \tag{3.14}$$

$$\sum_{f=1}^{\mathcal{F}} \sum_{i=1}^{df-1} w_{fi} = 1, \quad \sum_{f=1}^{\mathcal{F}} \sum_{i=1}^{df-1} m_{fi} = 1 \tag{3.15}$$

where $\mathbf{M}$ is a large positive number.

The observations are described with $\mathcal{F}$ factors, and each factor is assumed to have df different levels. The number of levels for a specific factor is defined with respect to the values observed in the training set. Note that, for an observation, as the value of a factor takes higher values, it is assumed that, it has a greater potential to be classified as positive. Therefore, the relation between the factor values and the given classification of an observation in training set is represented with a monotonically increasing function. We also call these factor values (w_{fi} and m_{fi}) as incremental variables.

The model aims to develop a set of additive utility functions, such that, the values of the decision variables (w_{fi} and m_{fi}) are consistent with the objective of the model. In Constraint sets (3.9 - 3.10), an error term e is used and when $e(p)$ takes a positive value, then the observation p is noted as misclassified. The error term $e(p)$ is equal to zero, if the observation p is classified correctly. These constraint sets can also be given as follows:

$$U(p) - \widetilde{U}(p) + e(p) \geq s \quad \forall p \in \mathcal{S}^+, j : x_f(p) = o_{fj} \in O_f \tag{3.16}$$

$$\widetilde{U}(p) - U(p) + e(p) \geq s \quad \forall p \in \mathcal{S}^-, j : x_f(p) = o_{fj} \in O_f \tag{3.17}$$

where $U(p)$ and $\widetilde{U}(p)$ are utility functions of observation p for positive and negative classifications, respectively.

For a misclassified observation $p \in \mathcal{S}^+$, when $U(p) \leq \widetilde{U}(p)$, the error term is equal to $\widetilde{U}(p) - U(p) + s$. Similarly, when $\widetilde{U}(p) \leq U(p)$, for this misclassified observation $p \in \mathcal{S}^-$, the error term is equal to $U(p) - \widetilde{U}(p) + s$. Note that, a small positive constant s ensures strict inequality.

For a misclassified observation with a positive error term ($e(p)$), corresponding indicator variable $Ind[p]$ is equal to 1. Constraint (3.12) forces that at most L number of observations in set $\mathcal{S}^-$ can be misclassified. In other words, it ensures to keep false positive classification error below a certain level, while objective function minimizes the number of false negative classification error.

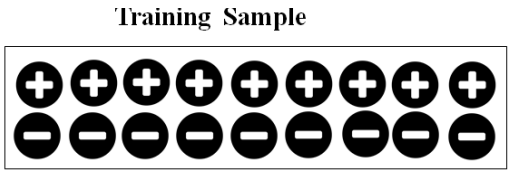
The non-negativity and monotonicity requirements of the model are ensured in Constraints (3.13) and (3.14), respectively. The Constraint (3.15) normalizes the global utilities in the interval [0,1].

It is important to note that, the proposed model is able to deal with categorical, discrete and continuous predictor (independent) variables.

In short, with respect to the given classifications of patients in the training sample, PCM(L) finds the optimal values for the incremental variables (w_{fi} and m_{fi}). Its objective is minimizing number of false negative classification while keeping the number of false positive classification below the specified level L. Solving PCM

for different values of parameter L results in solutions with different number of false positive and false negative classifications. For bigger values of the parameter L, PCM allows greater amount of false positive misclassification and achieves smaller false negative classification. Thus, for each L value, PCM finds different optimal values of decision variables, w_{fi} and m_{fi} and the initial populations of evolutionary algorithms are obtained by these different L levels.

The Figure 3.3 describes the sensitivity and specificity values of the solutions obtained by solving PCM with different values of L. Assume the training set consists of 10 patients with and 10 patients without the disease. Since the objective function minimizes the number of false negative classifications and by means of a constraint, it is intended to keep the number of false negative classification under the given value of L, solving PCM with $L \in \{0, 1, ..., 10\}$ brings different number of false positive and false negative classifications. In the figure, these are represented with $|FP|$ and $|FN|$. The graph at the bottom of Figure 3.3 illustrates the two dimensional space of true positive and true negative responses that the solution set spread over the Pareto-optimal front.



Ten patients with the disease
Ten patients without the disease

	PCM(0) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 0$	PCM(1) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 1$	PCM(2) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 2$	PCM(3) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 3$	PCM(4) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 4$	PCM(5) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 5$	PCM(6) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 6$	PCM(7) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 7$	PCM(8) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 8$	PCM(9) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 9$	PCM(10) $\min \lvert FN\rvert$ $s.t$ $\lvert FP\rvert \leq 10$
	$\lvert FP\rvert = 0$ $\lvert FN\rvert = 10$	$\lvert FP\rvert = 1$ $\lvert FN\rvert = 8$	$\lvert FP\rvert = 2$ $\lvert FN\rvert = 6$	$\lvert FP\rvert = 3$ $\lvert FN\rvert = 5$	$\lvert FP\rvert = 4$ $\lvert FN\rvert = 3$	$\lvert FP\rvert = 5$ $\lvert FN\rvert = 2$	$\lvert FP\rvert = 6$ $\lvert FN\rvert = 0$	$\lvert FP\rvert = 6$ $\lvert FN\rvert = 0$	$\lvert FP\rvert = 6$ $\lvert FN\rvert = 0$	$\lvert FP\rvert = 6$ $\lvert FN\rvert = 0$	$\lvert FP\rvert = 6$ $\lvert FN\rvert = 0$
Sensitivity	0.00	0.20	0.40	0.50	0.70	0.90	1.00	1.00	1.00	1.00	1.00
Specificity	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.40	0.40	0.40	0.40

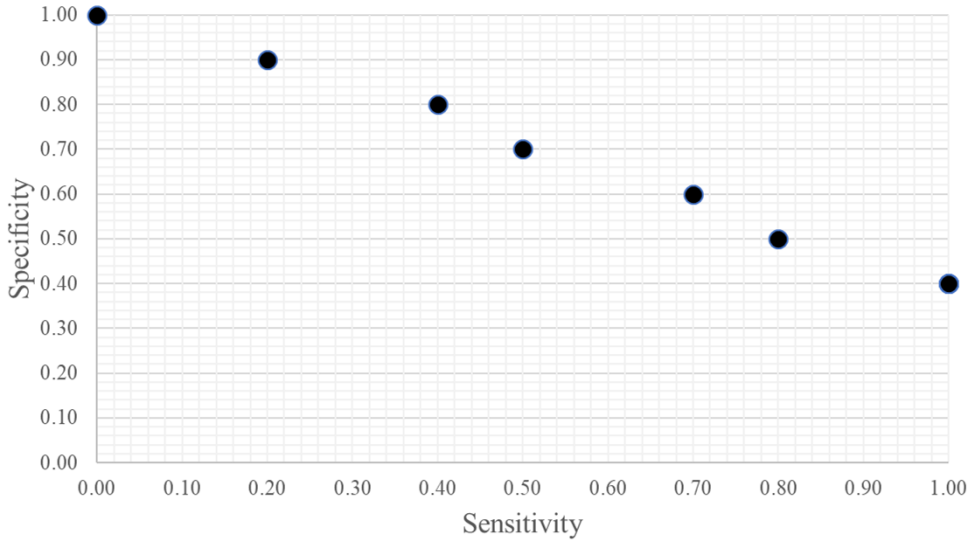


Figure 3.3: Example: Set of Solutions Obtained by PCM

3.2 Basic Characteristics of the Proposed Evolutionary Algorithms NSGA-II and RECGA

Each individual (W, M) pair in the initial population acquired by PCM, is a candidate parent to produce an offspring. Except the first generation, population size is set to the *PopulationSize* for each generation. Therefore, the number of solutions is increased up to the *PopulationSize* by crossover and mutation mechanisms. The crossover probability is 100% since both offspring and parent populations are carried to the next generation and the selection is made from all of these solutions. Real and linear crossover operations are used randomly.

We have designed genetic operators *RealCrossover* (Algorithm 3), *LinearCrossover* (Algorithm 4) and *Mutation* (Algorithm 5) in specific to our problem characteristics. Now, let us explain these operators with a small example. Consider a small instance having four factors ($\mathcal{F} = \{1, 2, 3, 4\}$), where they have two, four, three and five distinct values, respectively. Hence, the sizes of weight vectors (w_1, m_1), (w_2, m_2), (w_3, m_3), and (w_4, m_4) are one, three, two and four, respectively. Let (W^{p1}, M^{p1}) and (W^{p2}, M^{p2}) be two parent solutions and are given as follows:

$$W^{p1} = \begin{bmatrix} w_1 = & 0.20 & & & \\ w_2 = & 0.05 & 0.18 & 0.02 & \\ w_3 = & 0.09 & 0.03 & & \\ w_4 = & 0.10 & 0.17 & 0.14 & 0.2 \end{bmatrix} \quad M^{p1} = \begin{bmatrix} m_1 = & 0.01 & & & \\ m_2 = & 0.02 & 0.12 & 0.05 & \\ m_3 = & 0.19 & 0.08 & & \\ m_4 = & 0.13 & 0.10 & 0.23 & 0.07 \end{bmatrix}$$

$$W^{p2} = \begin{bmatrix} w_1 = & 0.03 & & & \\ w_2 = & 0.08 & 0.17 & 0.04 & \\ w_3 = & 0.09 & 0.14 & & \\ w_4 = & 0.07 & 0.12 & 0.15 & 0.11 \end{bmatrix} \quad M^{p2} = \begin{bmatrix} m_1 = & 0.12 & & & \\ m_2 = & 0.15 & 0.17 & 0.11 & \\ m_3 = & 0.04 & 0.04 & & \\ m_4 = & 0.07 & 0.05 & 0.17 & 0.08 \end{bmatrix}$$

Assuming the crossover point is between 3^{rd} and 4^{th} rows, the real crossover takes w_1, w_2 and w_3 from one parent and w_4 from the other parent for both of the weight vector sets W and M. The resulting offspring (W^{o1}, M^{o1}) is as follows:

$$W^{o1} = \begin{bmatrix} w_1 = & 0.20 & & & \\ w_2 = & 0.05 & 0.18 & 0.02 & \\ w_3 = & 0.09 & 0.03 & & \\ w_4 = & 0.07 & 0.12 & 0.15 & 0.11 \end{bmatrix} \quad M^{o1} = \begin{bmatrix} m_1 = & 0.01 & & & \\ m_2 = & 0.02 & 0.12 & 0.05 & \\ m_3 = & 0.19 & 0.08 & & \\ m_4 = & 0.07 & 0.05 & 0.17 & 0.08 \end{bmatrix}$$

The linear crossover mechanism produces a convex combination of parents using a randomly chosen multiplier, $\gamma \in (0, 1)$. If we employ the same parents (W^{p1}, M^{p1}) and (W^{p2}, M^{p2}), the vectors of the child are given as follows:

$$W^{o2} = \begin{bmatrix} w_1 = & \gamma 0.20 + (1-\gamma)0.03 & & & \\ w_2 = & \gamma 0.05 + (1-\gamma)0.08 & \gamma 0.18 + (1-\gamma)0.17 & \gamma 0.02 + (1-\gamma)0.04 & \\ w_3 = & \gamma 0.09 + (1-\gamma)0.09 & \gamma 0.03 + (1-\gamma)0.14 & & \\ w_4 = & \gamma 0.10 + (1-\gamma)0.07 & \gamma 0.17 + (1-\gamma)0.12 & \gamma 0.14 + (1-\gamma)0.15 & \gamma 0.2 + (1-\gamma)0.11 \end{bmatrix}$$

$$M^{o2} = \begin{bmatrix} m_1 = & \gamma 0.01 + (1-\gamma)0.12 & & & \\ m_2 = & \gamma 0.02 + (1-\gamma)0.15 & \gamma 0.12 + (1-\gamma)0.17 & \gamma 0.05 + (1-\gamma)0.11 & \\ m_3 = & \gamma 0.19 + (1-\gamma)0.04 & \gamma 0.08 + (1-\gamma)0.04 & & \\ m_4 = & \gamma 0.13 + (1-\gamma)0.07 & \gamma 0.10 + (1-\gamma)0.05 & \gamma 0.23 + (1-\gamma)0.17 & \gamma 0.07 + (1-\gamma)0.08 \end{bmatrix}$$

The mutation operator randomly chooses a factor and changes its weights by adding a randomly chosen value δ from a given continuous interval. Note that, since w_{fi} and m_{fi} are allowed to take only positive values, if any of them is changed to a negative value, it is reset to the small positive constant, t. Assume that the second factor is chosen ($k = 2$) and let δ chosen from the range $[-0.5, 0.5]$ be equal to -0.4. If we apply the mutation operator on (W^{o1}, M^{o1}), then it will be changed to the following mutated individual.

$$W^{o3} = \begin{bmatrix} w_1 = & 0.20 & & & \\ w_2 = & 0.01 & 0.14 & t & \\ w_3 = & 0.09 & 0.03 & & \\ w_4 = & 0.07 & 0.12 & 0.15 & 0.11 \end{bmatrix} \quad M^{o3} = \begin{bmatrix} m_1 = & 0.01 & & & \\ m_2 = & t & 0.08 & 0.01 & \\ m_3 = & 0.19 & 0.08 & & \\ m_4 = & 0.07 & 0.05 & 0.17 & 0.08 \end{bmatrix}$$

Either it is mutated or not, a new offspring should be normalized before it is added to the population. In other words, the equalities $\sum_{f=1}^{\mathcal{F}} \sum_{i=1}^{df} w_{fi} = 1$ and $\sum_{f=1}^{\mathcal{F}} \sum_{i=1}^{df} m_{fi} = 1$ must hold. Algorithms 3 - 6, illustrate real crossover, linear crossover, mutation and normalization procedures, respectively.

Algorithm 3 *RealCrossover*

for $f \in \mathcal{F}$ **do**
    for $i \in \{1, 2, \ldots, df - 1\}$ **do**
        $w_{fi} = 0, m_{fi} = 0$
    end for
end for
Randomly choose a crossover point cp from the set $\{1, 2, \ldots, F\}$
for $f \in \{1, 2, \ldots, F\}$ **do**
    if $f \leq cp$ **then**
        for $i \in \{1, 2, \ldots, df - 1\}$ **do**
            $w_{fi} = w^1_{fi}, m_{fi} = m^1_{fi}$
        end for
    else
        for $i \in \{1, 2, \ldots, df - 1\}$ **do**
            $w_{fi} = w^2_{fi}, m_{fi} = m^2_{fi}$
        end for
    end if
end for
$Normalize$ the weights of (W, M)

Algorithm 4 *LinearCrossover*

for $f \in \mathcal{F}$ **do**
  for $i \in \{1, 2, \ldots, df - 1\}$ **do**
    $w_{fi} = 0, m_{fi} = 0$
  end for
end for
Randomly choose a multiplier $\gamma \in [0, 1]$
for $f \in \{1, 2, \ldots, F\}$ **do**
  for $i \in \{1, 2, \ldots, df - 1\}$ **do**
    $w_{fi} = \gamma w^1_{fi} + (1 - \gamma) w^2_{fi}$
    $m_{fi} = \gamma m^1_{fi} + (1 - \gamma) m^2_{fi}$
  end for
end for
$Normalize$ the weights of (W, M)

Algorithm 5 *Mutation*

Randomly choose a mutation point k from the set $\{1, 2, \ldots, F\}$,
for $i \in \{1, 2, \ldots, df - 1\}$ **do**
  Randomly choose a number $\gamma \in [-0.5, 0.5]$,
  $w_{ki} = \max\{w_{ki} + \gamma, t\}$,
  $m_{ki} = \max\{m_{ki} + \gamma, t\}$,
end for
$Normalize$ the weights of (W, M)

Algorithm 6 *Normalize*

Input: A solution (W, M),
Let $sumw = 0$ and $summ = 0$,
for $f \in \mathcal{F}$ **do**
    for $i \in \{1, 2, \ldots, df - 1\}$ **do**
        $sumw \leftarrow sumw + w_{fi}$
        $summ \leftarrow summ + m_{fi}$
    end for
end for
for $f \in \mathcal{F}$ **do**
    for $i \in \{1, 2, \ldots, df - 1\}$ **do**
        $w_{fi} \leftarrow w_{fi}/sumw$
        $m_{fi} \leftarrow m_{fi}/summ$
    end for
end for
Output: Normalized solution (W, M).

In the following section we describe the use of genetic operators in the evolutionary algorithm NSGA-II and we also explain PCM+NSGA-II as a classification algorithm.

3.3 PCM+NSGA-II

NSGA-II algorithm starts with a random initial parent population and by performing crossover and mutation operations on randomly chosen two parents from this set, an offspring is created. We refer this procedure as *GenerateOffspring* (Algorithm 7). The offspring generation procedure iterates until the union of existing and new solutions, R_t, reaches to a predetermined size. Through non-dominated sorting approach (referred as *fast-non-dominated-sort* in [4]) each solution in set R_t is assigned to a front. Solutions belonging to the first front Fr_1, are the non-dominated solutions of the set R_t. Once these solutions are found, they are set apart. Non-dominated solutions of the remaining set assigned to second front, Fr_2. This procedure continues until all front members are found. To compose the set of solutions to be carried to the

next generation, individuals are selected starting with the first front, where the solutions with lower (better) rank are preferred. This procedure is followed until there is no more front whose all members can be carried to the next generation without exceeding predetermined population size. At this point, if the selected solutions did not reach the population size, the solutions of the next front is sorted by their crowding distances which are based on *crowding-distance-assignment* in [4]. Then the solutions are selected based on crowding comparison approach which favors the solutions located in less crowded regions. The solutions are selected according to this method as members of next generation, to preserve diversity of solutions.

Once a population is formed, selection, crossover and mutation mechanisms are performed to create the new population and the procedure iterates in a similar manner until a termination criterion is satisfied.

As previously mentioned, the initial population of NSGA-II algorithm is obtained by solving PCM for different values of parameter L. The aim of employing NSGA-II is to achieve a better population (solutions) in terms of both sensitivity and specificity.

To measure the quality of new solutions, (w, m) values are utilized to identify the classes of the patients in a sample, $\mathcal{V}$, which has no common members with the training set, $\mathcal{S}$. Performance of a solution is evaluated through its sensitivity and specificity values. Namely, we validate the solutions in a generation by using a dataset of new members that are not used in the training set that we have utilized before, in order to avoid overfitting. Then, a selected subset of solutions (of a size that is at most equal to *GenerationSize*) is carried to the next generation. Selection of solutions is made by utilizing *fast-non-dominated- sort* and *crowding-distance-assignment* operations.

In non-domination ranking, the criteria are sensitivity and specificity. Assume that, (W, M) is a solution and $sensitivity_{(W,M)}$ and $specificity_{(W,M)}$ are true positive and true negative rates obtained by this solution, respectively. Then, (W^1, M^1) $dominates$ (W^2, M^2) if

$$sensitivity_{(W^1,M^1)} \geq sensitivity_{(W^2,M^2)}$$

and

$$specificity_{(W^1,M^1)} > specificity_{(W^2,M^2)}$$

or

$$sensitivity_{(W^1,M^1)} > sensitivity_{(W^2,M^2)}$$

and

$$specificity_{(W^1,M^1)} \geq specificity_{(W^2,M^2)}.$$

In this case, (W, M) is called as non-dominated if no other solution dominates it. Figure 3.4 illustrates the flow chart of PCM+NSGA-II.

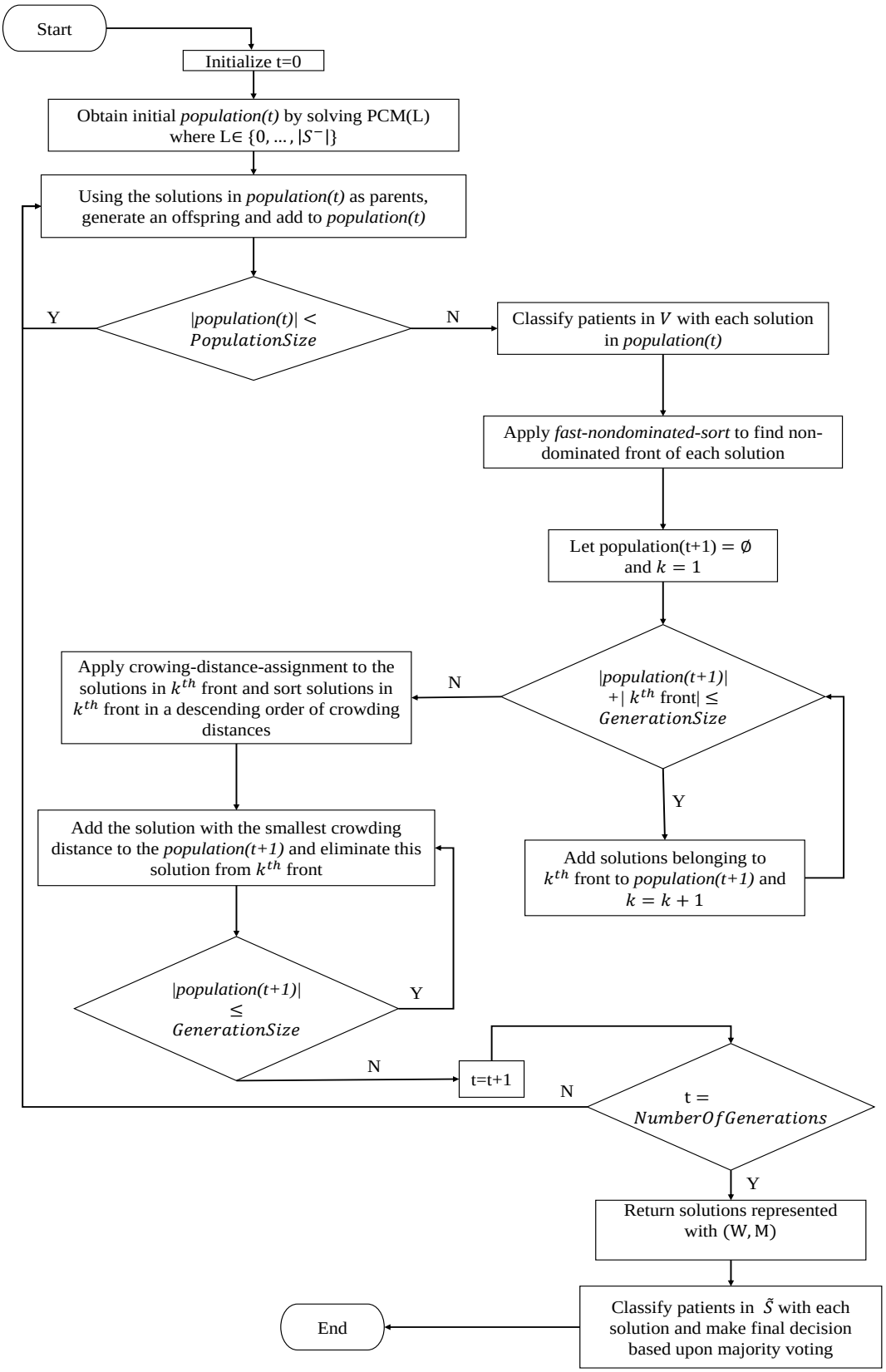


Figure 3.4: Flow Chart of PCM+NSGA-II

In our problem setting, there are finitely many different values of sensitivity and specificity pairs. Therefore, there may exist many solutions which have the same sensitivity and specificity values. Thus, as the number of generations increases, the diversity of solutions decreases. To preserve diversity, we keep one solution from the set of solutions which have the same sensitivity and specificity values. Otherwise, as the generation number increases the diversity is lost. Besides that, the crowding distance operator is another diversity preservation mechanism of the algorithm. We encourage the preservation of diversity in each iteration to have a greater chance of finding better solutions at the end of the algorithm.

When the algorithm terminates, we obtain a set of solutions which is comprised of the variables (w, m). Then, the classification of a new patient from the test sample is determined with the same methodology that is used before (see Algorithm 1), where the incremental values for the patients in test sample, $\widetilde{\mathcal{S}}$, are found by linear interpolation. Their global utilities, U and $\widetilde{U}$ are calculated. Each solution makes a decision by comparing these global utilities. The counters that counts number of votes for positive and negative prediction for a patient are kept and the final classification of a new patient is determined by comparing these counter values, based upon majority voting.

In a nutshell, first, PCM provides the initial parameters and then the evolutionary algorithm NSGA-II tunes these parameter values using a separate set. Finally, the resulting solutions are tested with a test set to measure the performance of the model. Prediction procedure with PCM +NSGA-II is given in Algorithm 8. The computer code of PCM+NSGA-II is available in [95].

Like PCM, PCM+NSGA-II is also able to deal with categorical, discrete and continuous predictor (independent) variables.

Algorithm 7 $GenerateOffspring(P_t)$

1: Randomly choose two parents $(W^1, M^1) \in P_t$ and $(W^2, M^2) \in P_t$
2: Perform either a ***RealCrossover*** or ***LinearCrossover*** with probabilities p_{rc} and p_{lc}, respectively to obtain a new offspring (W^3, M^3)
3: Perform ***mutation*** on (W^3, M^3) with probability p_m

Algorithm 8 Prediction with PCM+NSGA-II

for each $p \in \widetilde{\mathcal{S}}$ **do**
    Initialize the counter for positive and negative predictions $CP(p) = 0$ and $CN(p) = 0$, respectively.
end for
Initialize $t = 0$ and $P_t = (wset, mset) = \emptyset$
for each $L \in \{0, 1, \ldots, |S^-|\}$ **do**
    Solve $PCM(L)$ and let $((W^*(L), M^*(L))$ be an optimal solution,
    $P_t \leftarrow P_t \cup \{(W^*(L), M^*(L))\}$
end for
while $t < NumberOfGenerations$ **do**
    $Q_t \leftarrow \emptyset$
    while $|P_t| + |Q_t| < PopulationSize$ **do**
        $(W', M') \leftarrow GenerateOffspring(P_t \cup Q_t)$
        if All weights (w_{fi} and m_{fi}) of offspring $(W', M') > 0$ **then**
            $Q_t \leftarrow Q_t \bigcup \{(W', M')\}$
        end if
    end while
    $R_t = P_t \cup Q_t$
    for each $(W^i, M^i) \in R_t$ **do**
        Initialize $sensitivity(i) = specificity(i) = 0$
        Calculate utility functions $u_f(\cdot)$ and $\widetilde{u}_f(\cdot)$ for each factor using the procedure *GenerateUtilityFunctions*,
        for $p \in \mathcal{V}$ **do**
            Set total utility value $U(p) = 0$ and total disutility value $\widetilde{U}(p) = 0$,
            for each $f \in \mathcal{F}$ **do**
                $U(p) \leftarrow U(p) + u_f(x_f(p))$,
                $\widetilde{U}(p) \leftarrow \widetilde{U}(p) + \widetilde{u}_f(x_f(p))$,
            end for
            if $U(p) \geq \widetilde{U}(p)$ **then**
                $\widetilde{y}(p) = 1$
            else
                $\widetilde{y}(p) = 0$
            end if
        end for
        Let $sensitivity(i) = \frac{\text{number of obervation } p \text{ in } V^+ \text{ with } \widetilde{y}(p)=1}{|V^+|}$,
        $specificity(i) = \frac{\text{number of obervation } p \text{ in } V^- \text{ with } \widetilde{y}(p)=0}{|V^-|}$
        $Fr =$ *fast-non-dominated-sort* (R_t)
        Eliminate duplicate solutions in the same front in Fr
        $P_{t+1} = \emptyset$ and $k = 1$

Algorithm 8 continues

if $|P_{t+1}| + |Fr_k| \leq GenerationSize$ **then**
  $P_{t+1} = P_{t+1} \cup Fr_k$
  $k = k + 1$
end if
Apply *crowding-distance-assignment*(Fr_k)
Sort solutions in Fr_k in a descending order of crowding distances
$P_{t+1} = P_{t+1} \cup Fr_k[1 : GenerationSize - |P_{t+1}|)]$
$Q_{t+1} = GenerateOffspring(P_{t+1})$
$t = t + 1$
end for
end while
for each $(W, M) \in P_{NumberOfGenerations}$ **do**
Apply *GenerateUtilityFunctions* to calculate $u_f(\cdot)$ and $\widetilde{u}_f(\cdot)$
for $p \in \widetilde{\mathcal{S}}$ **do**
$U(p) := 0$ and $\widetilde{U}(p) := 0$
for each $f \in \mathcal{F}$ **do**
$U(p) \leftarrow U(p) + u_f(x_f(p))$,
$\widetilde{U}(p) \leftarrow \widetilde{U}(p) + \widetilde{u}_f(x_f(p))$
end for
if $U(p) \geq \widetilde{U}(p)$ **then**
$CP(p) \leftarrow CP(p) + 1$
else
$CN(p) \leftarrow CN(p) + 1$
end if
end for
end for
for $p \in \widetilde{\mathcal{S}}$ **do**
if $CP(p) \geq CN(p)$ **then**
$\widetilde{y}(p) = 1$
else
$\widetilde{y}(p) = 0$
end if
end for

3.4 PCM+RECGA

Since a model can easily achieve high overall prediction rates just by assigning all the observations to the class of the majority, for problems where the number of positive observations is significantly smaller than the number of negative observations, achieving high true positive and true negative predictions simultaneously is more important. To detect the presence or absence of a disease and develop a robust rare event classification algorithm for medical usage, we combine PCM with another evolutionary algorithm, Rare Event Classifier Genetic Algorithm (RECGA).

The RECGA starts with an initial solution set $\mathcal{X}$. The algorithm starts with randomly choosing two parents from $\mathcal{X}$ and it performs either a real crossover or linear crossover according to their probabilities of selection p_{rc} and p_{lc}, respectively. Then, the offspring is subject to mutation with a given mutation probability p_m. This process is repeated until the number of solutions in the current generation reaches to the *PopulationSize*. The solutions are evaluated by the fitness function, Fscore. Fitness function of a solution is calculated due to its classification performance in set $\mathcal{V}$. The set $\mathcal{V}$ has no members from $\mathcal{S}$, which is utilized by PCM as the training sample. Until the number of solutions, whose fitness value is greater than a specific $threshold$ value, become greater than or equal to *minFinalSetSize*, the algorithm continues to produce solutions via genetic operations.

Note that, the fitness value (Fscore) of a solution is calculated as a combination of its classification ability in both classes under consideration. Thus, fitness function ensures that, the classification of rare events is as important as the classification of frequent events.

The RECGA is expressed in Algorithm 9 which utilizes the *RealCrossover*, *LinearCrossover*, and *Mutation* genetic operations given in Section 3.3.

Algorithm 9 *RareEventClassifierGeneticAlgorithm (RECGA)*

Let $\mathcal{X}$ be an initial solution set.
continue $\leftarrow$ true
while continue=true **do**
  while $|\mathcal{X}| < PopulationSize$ **do**
    $(W', M') \leftarrow GenerateOffspring(\mathcal{X})$
    if All weights (w_{fi} and m_{fi}) of offspring $(W', M') > 0$ **then**
      $\mathcal{X} \leftarrow \mathcal{X} \cup \{(W^3, M^3)\}$
    end if
  end while
  for each $(W^i, M^i) \in \mathcal{X}$ **do**
    Initialize $sensitivity(i) = specificity(i) = Fscore(i) = 0$
    Calculate utility functions $u_f(.)$ and $\widetilde{u}_f(.)$ for each factor using the procedure *GenerateUtilityFunctions*
    for $p \in \mathcal{V}$ **do**
      Set $U(p) := 0$, $\widetilde{U}(p) := 0$
      for each $f \in \mathcal{F}$ **do**
        $U(p) \leftarrow U(p) + u_f(x_f(p))$
        $\widetilde{U}(p) \leftarrow \widetilde{U}(p) + \widetilde{u}_f(x_f(p))$
      end for
      if $U(p) \geq \widetilde{U}(p)$ **then**
        $\widetilde{y}(p) = 1$
      else
        $\widetilde{y}(p) = 0$
      end if
    end for
    Let $sensitivity(i) = \frac{\text{number of obervation } p \text{ in } V^+ \text{ with } \widetilde{y}(p)=1}{|V^+|}$,
    $specificity(i) = \frac{\text{number of obervation } p \text{ in } V^- \text{ with } \widetilde{y}(p)=0}{|V^-|}$
    $Fscore(i) = \frac{2 \times sensitivity(i) \times specificty(i)}{sensitivity(i) + specificity(i)}$
    newSet $\leftarrow \emptyset$, $threshold \leftarrow 1$

Algorithm 9 continues

```
        while newSet = ∅ and threshold ≥ 0 do
            for each (W^i, M^i) ∈ X do
                if Fscore(i) ≥ threshold then
                    newSet ← newSet ∪{(W^i, M^i)}
                end if
            end for
            threshold ← threshold - 0.1
        end while
        X ← ∅
        X ← newSet
        if |X| ≥ minFinalSetSize then
            continue ← false
        end if
    end for
end while
```

Like PCM+NSGA-II, we propose PCM+RECGA as a two-phase algorithm. In the first phase, PCM provides diverse initial solutions by changing the levels of parameter L. In the second phase, RECGA utilizes this diverse set of initial solutions to create offspring solutions via genetic operations until the number of solutions equal to the *PopulationSize*. Each solution on hand are used to classify the observations in a validation set, $\mathcal{V}$ that has no common members with $\mathcal{S}$, to avoid overfitting. Each solution has a sensitivity and specificity rate. The fitness function, Fscore, of each solution is calculated through these measures as previously explained. Only the solutions whose fitness value exceeds a *threshold* value are carried to the next generation. If there is no such solution, *threshold* value is decreased by a certain amount until a number of solutions are found that satisfy the criteria. Until the number of solutions reaches the predetermined *minFinalSetSize*, the algorithm continues to create new offspring. By this way, the algorithm is able to create solutions that have high fitness values. Thus, the selected solutions are likely to achieve high sensitivity and specificity, simultaneously.

PCM+RECGA is able to deal with categorical, discrete and continuous predictor (independent) variables. Figure 3.5 illustrates the flow chart of PCM+RECGA.

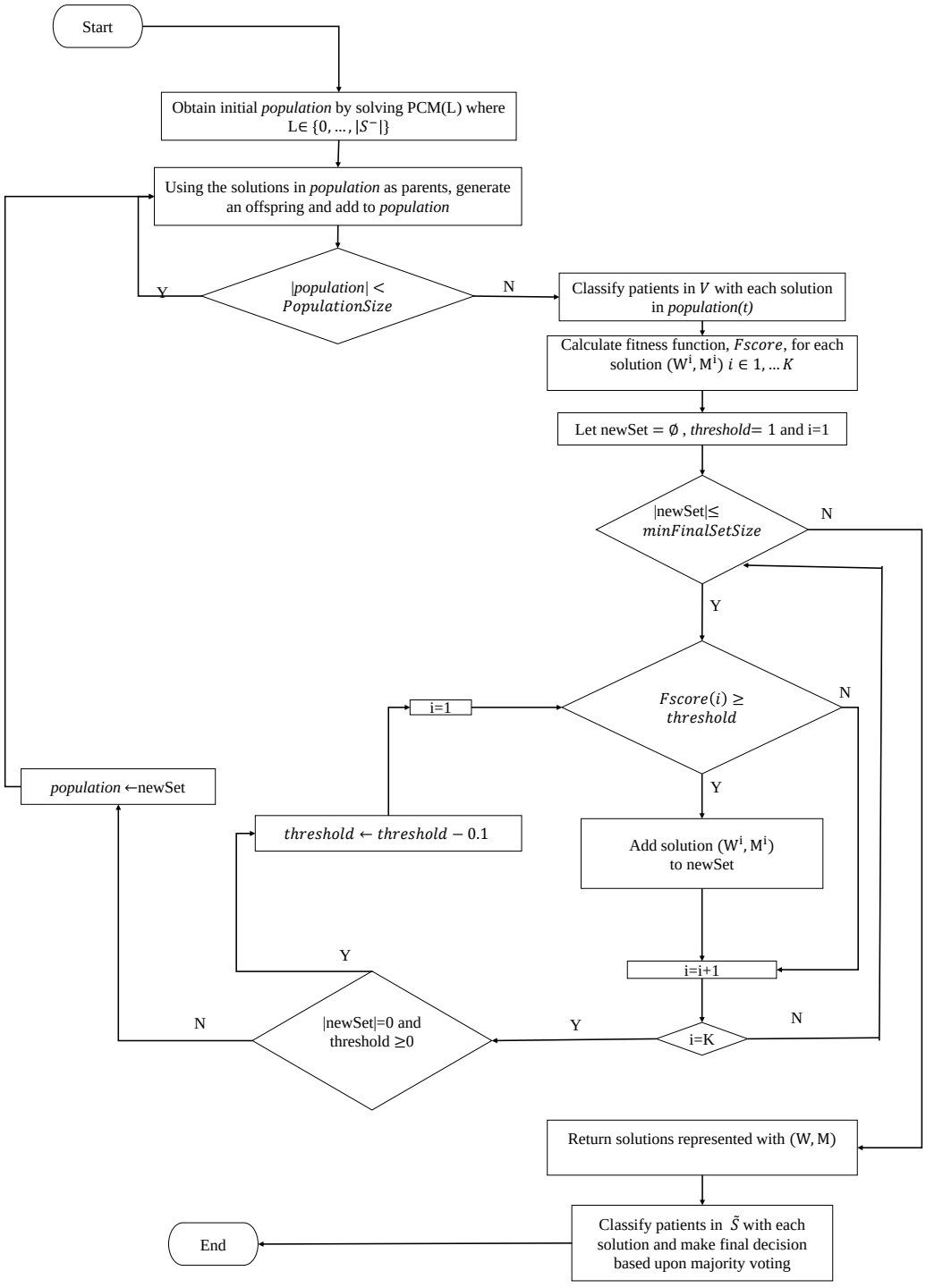


Figure 3.5: Flow Chart of PCM+RECGA

Algorithm 10 Prediction with PCM+RECGA

for each $p \in \widetilde{\mathcal{S}}$ **do**
    $CP(p) := 0$; $CN(p) := 0$
end for
$\mathcal{X}' := \emptyset$
for each $L \in \{0, 1, \ldots, |S^-|\}$ **do**
    Solve $PCM(L)$
    $\mathcal{X}' \leftarrow \mathcal{X}' \cup \{(W^*(L), M^*(L))\}$
end for
Apply *RareEventClassifierGeneticAlgorithm (RECGA)* to $\mathcal{X}'$ to obtain $\mathcal{X}$
for each $(W, M) \in \mathcal{X}$ **do**
    Apply *GenerateUtilityFunctions* to calculate $u_f(\cdot)$ and $\widetilde{u}_f(\cdot)$
    for $p \in \widetilde{\mathcal{S}}$ **do**
        $U(p) := 0$ and $\widetilde{U}(p) := 0$
        for each $f \in \mathcal{F}$ **do**
            $U(p) \leftarrow U(p) + u_f(x_f(p))$,
            $\widetilde{U}(p) \leftarrow \widetilde{U}(p) + \widetilde{u}_f(x_f(p))$
        end for
        if $U(p) \geq \widetilde{U}(p)$ **then**
            $CP(p) \leftarrow CP(p) + 1$
        else
            $CN(p) \leftarrow CN(p) + 1$
        end if
    end for
end for
for $p \in \widetilde{\mathcal{S}}$ **do**
    if $CP(p) \geq CN(p)$ **then**
        $\widetilde{y}(p) = 1$
    else
        $\widetilde{y}(p) = 0$
    end if
end for

Once PCM+RECGA terminates with a final set of solutions, they are used to classify patients in the test set, $\widetilde{\mathcal{S}}$. Note that, $\widetilde{\mathcal{S}}$, $\mathcal{S}$ and $\mathcal{V}$ are mutually exclusive sets. Algorithm 10 gives the outline of PCM+RECGA and how PCM+RECGA works is explained with a simple example in the Appendix (Section A).

3.5 Hyper-parameter Optimization for PCM+NSGA-II and PCM+RECGA

For the suggested models, there are some hyper-parameters whose values must be specified externally by the user. The hyper-parameters of each model/algorithm are listed in Table 3.7.

Table 3.7: Hyper-parameters of the Models/Algorithms

PCM	**NSGA-II**	**RECGA**
$t = 10^{-16}$	$PopulationSize$	$PopulationSize$
$s = 0.0001$	$GenerationSize$	$minFinalSetSize$
	$NumberOfGenerations$	p_{rc}, p_{lc}
	p_{rc}, p_{lc}	p_m
	p_m	

Among the given hyper-parameters, t is related with the monotonically increasing assumption of PCM. Besides it ensures that incremental variables should take values greater than zero, it also provides a lower bound for the difference between the incremental variables of the consecutive factor levels. Since finding contributions of each factor level to the utility functions is the main concern of the model, we choose a very small value of t in our computational experiments, to minimize its effect on the values of incremental variables. Hyper-parameter s is used in the first two constraints of PCM and it ensures strict inequality between global utility pairs. We use a relatively higher value of s compared to t, to avoid an erroneous classification decision when the difference between U and $\widetilde{U}$ is small. The values of these two hyper-parameters are chosen as indicated in Table 3.7.

To find the optimal values of the remaining hyper-parameters, we applied hyper-parameter optimization. In this process, we run the model with different combina-

tions of the values of hyper-parameters, and select the best combination according to models' classification performances. It is important to note that, hyper-parameter values should be determined by an unbiased estimate of the generalization performance of the model [96]. In accordance with this purpose, we apply nested cross validation, where it is referred as a common approach for hyper-parameter optimization [97], [98], [99].

Nested cross validation is defined as two nested loops of cross validation. Due to the inner cross validation performance, the hyper-parameter values are set and the outer loop evaluates the generalization ability of the model with the selected values of hyper-parameters on an independent set of observations [96], [97]. By this way, nested cross validation ensures that, the model do not use the observations reserved for outer loop to tune the hyper-parameters. The detailed explanation of hyper-parameter tuning process can be found in the Appendix (Section B). Note that, the hyper-parameter optimization process is repeated for each model and for each dataset.

In the following chapter, we discuss the application of PCM+NSGA-II and PCM+RECGA in the problem of classification of patients due to the risk of restenosis after coronary stent implantation.

CHAPTER 4

PATIENT CLASSIFICATION CONSIDERING THE RISK OF RESTENOSIS AFTER CORONARY STENT IMPLANTATION

4.1 Coronary In-Stent-Restenosis

As a result of aging and due to some life style habits, plaque accumulates in the blood vessels of the heart. This accumulation causes narrowing of these vessels and impedes the flow of the blood through them. This narrowing of the arteries is called as atherosclerosis [100]. For patients who suffer from the narrowing of coronary arteries, namely coronary heart disease patients; stents are used as one of the main therapeutic procedures [101]. By a balloon catheter, the stent is inserted into the clogged artery, and with the inflation of the balloon, the stent, which is a "tiny wire mesh tube", expands to open the artery. Stent stays in this artery permanently to keep it open in order to ensure the flow of blood and reduce the risk of heart attack [102].

There is re-narrowing risk of arteries after the balloon angioplasty and other procedures that use catheters. Although the acute operation success of stents is very high and usage of stents reduces this risk [102], the possibility of re-narrowing (in-stent-restenosis) in the following period is still an issue.

"Restenosis is defined as a section of blocked artery that was opened up with an angioplasty or a stent narrowed again" [103]. From a more medical point of view, it is defined as "either a luminal narrowing of at least 50% of the vessel diameter with associated evidence of functional significance by symptoms of ischemia or abnormal fractional flow reserve, or luminal narrowing of at least 70% or greater in the absence of ischemic symptoms" [104].

It is important to note that, narrowing of arteries in the case of restenosis is not caused by plaques, differently from the case of atherosclerosis. Restenosis is a recovery response of the stented artery, where the stent implantation traumatized the surface of the relevant vessel [105].

Our focus in this chapter is in-stent-restenosis, where re-narrowing of the artery occurs after the stent implantation. In-stent-restenosis usually occurs within the first six months after the initial procedure and "symptoms are very similar to the symptoms that initially brought the patients to the interventional cardiologist". It is detected via the follow-ups conducted by the medical expert [106]. When it is detected, the main objective is to open the relevant artery before the patient has heart attack and gets irreversible myocardial tissue damage. Therefore, it has an utmost importance to foresee the restenosis risk of the patient. After investigating the relevant literature and making interviews with experts, potential predictors of in-stent-restenosis are identified as given in Table 4.1. The main determinants of the process leading to in-stent-restenosis can be categorized as factors related to the patient, disease, procedure and lesion. Time passed after the stent implantation is not included in the predictors given in Table 4.1 since it is known that the risk of observing in-stent-restenosis does not change with time.

The objective of this chapter is to classify patients according to their in-stent-restenosis risk without performing selective coronary angiography, which is an accurate, but also an expensive and risky procedure. For this purpose, we utilize patient, disease, procedure and lesion related predictors listed in Table 4.1.

Table 4.1: In-Stent-Restenosis Predictors

Patient	Disease	Procedure	Lesion
Diabetes mellitus [107, 105, 108]	Single vessel disease [108]	Multivessel stenting [107, 105]	Lesion length [108, 105]
Hypertension [107, 105]	Multivessel disease [108]	Type of stent [107, 105]	Left main coronary artery lesion [105]
Hyperlipidemia [109]	Extent of coronary disease [107]	Final minimal lumen diameter [108, 105]	Ostial lesion [105]
Smoking [107]	Clinical presentation with MI [108]	Stent size [107]	Calcific lesion [105]
Age [107]		Length of stent [105, 107, 108]	Lesions with complex morphology [108]
Sex [107, 108]		Initial diameter stenosis [108]	Chronic total occlusions [108]
Patient compliance [109]		Final diameter stenosis [108, 105]	
Prior PTCA [107]		Maximal balloon inflation pressure [108]	
Prior MI [107]		Elective or bailout implantation [107]	
Prior CABG [107, 108]		Target vessel: native coronary or SVG [107]	
Family history [109]		Existence of full revascularization [109]	
Body mass index [109]		Post procedure complications [109]	
End stage renal disease [109]		Post treatment with anticoagulation	
on hemodialysis [109]		and/or antiplatelet drugs [107]	
Left ventricular function			
before stenting [107]			

The other tools that doctors can use in diagnosis of in-stent-restenosis are clinical evaluation, exercise stress test, computerized tomography angiography, myocardial perfusion scintigraphy and stress echocardiography. Clinically, one or more of these methods are used to detect the status of a patient. Andersen et al. claim that sensitivity and specificity of clinical evaluation are 26% and 84%, respectively [110]. Dori et al., Andersen et al. and Garzon and Eisenberg analyze the ability of exercise stress test in detection of in-stent-restenosis. Sensitivity of the method ranges from 26% to 54% and its specificity is between 70% and 77% [111, 110, 112]. Kósa et al. consider the exercise stress test and clinical evaluation together, which has a sensitivity between 21% and 26% a specificity between 68% and 86% [113]. Yang et al., Carrabba et al. and Gaspar et al. discuss the success of computerized tomography in detection of in-stent-restenosis. Sensitivity and specificity range from 86% to 89% and from 81% to 93%, respectively [114, 115, 116]. Elhendy et al., Dori et al., Kósa et al. and Garzon and Eisenberg give sensitivity and specificity of myocardial perfusion scintigraphy. These rates take values between 79% and 87% and between 78% and 83%, respectively [117, 111, 113, 112]. Finally, Dori et al. and Garzon and Eisenberg analyze the performance of stress echocardiography [111, 112]. Its sensitivity ranges from 63% to 82% where its specificity is about 87%.

These methodologies can be utilized only if the patient developed symptoms of the disease. However, some patients may not show any apparent symptoms or clinical findings of the restenosis. Therefore, a methodology that can classify the patients before the symptoms of the disease is developed helps to protect these patients from experiencing an emergency case, using more medication than necessary and losing the benefit of the treatment in the long run.

4.2 Data

After investigating the relevant literature and making interviews with the experts, among the 37 potential in-stent restenosis predictors listed in Table 4.1, we have decided to focus on 22 predictors given in Table 4.2. As claimed by Koller and Sahami [118] and as expected, reducing the number of irrelevant features reduces the running time of the algorithm. Blumer et al. indicate that, "given two explanations of the

data, all other things being equal, the simpler explanation is preferable." This principle is called as Occam's Razor and implies building the simplest model is better [119]. In this regard, random forest, stepwise regression, Boruta feature selection and LASSO (least absolute shrinkage and selection) feature selection methods are applied to increase the prediction ability by selecting the most relevant predictors of in-stent-restenosis. For this purpose, we create five different instances which are comprised of various subsets of all observations. Then we apply the given feature selection methods to these instances in order not to overfit to a specific instance. These methods are applied by the statistical software, R. Predictors that are found significant by each of these methods are marked in Table 4.2.

Table 4.2: Significant Predictors Due to Various Feature Selection Methodologies

	Age	Sex	Diabetes	**Chronic Renal Disease**	Hypertension	**Hyperlipidemia**	Smoking	BMI	Syntax Score	**Stent Type**	Complex Morphology	Bifurcation Lesion	**Prior PTCA**	**Prior MI**	Clinical Presentation With MI	**Stent Size**	Stent Length	Full Revascularization	Ostial Lesion	**Calcific Lesion**	**Native or SVG**	Prior CABG
Random Forest	✓	✓	✓			✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓				✓
	✓		✓		✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓			✓		
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓		✓		✓
		✓	✓		✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓		✓		✓
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓			✓		✓
Stepwise Regression						✓				✓	✓					✓				✓	✓	
				✓						✓				✓		✓				✓	✓	✓
			✓	✓	✓	✓			✓	✓			✓	✓		✓				✓	✓	✓
				✓		✓			✓	✓			✓	✓		✓				✓	✓	
				✓		✓				✓		✓	✓	✓		✓				✓	✓	
Boruta										✓										✓	✓	
										✓										✓	✓	
			✓	✓						✓				✓						✓	✓	
				✓						✓	✓	✓						✓		✓	✓	
				✓						✓				✓		✓				✓	✓	
LASSO										✓											✓	
										✓											✓	
										✓										✓	✓	
				✓						✓										✓	✓	
										✓										✓	✓	

After this preliminary analysis, we have determined that following 8 factors (shown bold in Table 4.2) are the most relevant predictors: stent type, existence of calcific lesion, existence of prior percutaneous transluminal coronary angioplasty (PTCA), existence of prior myocardial infarction (MI), stent size, existence of chronic renal disease, hyperlipidemia and target vessel.

The relevance of these factors with the disease can be explained from a medical point of view. Restenosis rate in drug-eluting stents compared to bare metal stents is 70-80% lower (25%-40% vs. 5%-10%). Therefore, the majority of stents used today are drug-eluting stents. The high calcification rate of the lesion adversely affects the stent placement in the vein and the development of restenosis. Patients who have experienced percutaneous transluminal coronary angiography and developed coronary restenosis have a higher rate of in-stent-restenosis due to neointimal proliferation in the same lesions, while patients with previous myocardial infarction have a weak relationship with in-stent-restenosis. Restenosis rate in patients with chronic renal disease is significantly higher than that of the patients without kidney disease and there is a negative correlation between stent diameter and restenosis. As the diameter increases, the chance of restenosis is reduced. The link between hyperlipidemia and in-stent-restenosis is relatively weak. Hyperlipidemia may accelerate atherosclerosis rather than restenosis, leading to the development of new lesions and restenosis is more likely to occur in saphenous vein grafts than in native veins [109].

Note that, the features selected by the feature selection methodologies may not exactly match the features found most relevant by medical professionals. For example, while the link between hyperlipidemia and stent-restenosis is found weak by experts, feature selection methods place hyperlipidemia among the most relevant factors. In this study, while we pay attention to the medical professionals' suggestions, we mainly rely on the statistical and mathematical methods in feature selection process.

The final set of predictors included in the models are given in Table 4.3. The second column of Table 4.3 indicates type of variables (categorical, integer, continuous) and the last column gives potential values. The response (dependent variable) is a binary variable which indicates whether a restenosis is expected to exist (1) or not (0) within the period of 36-month beginning with a coronary stent implantation.

Table 4.3: Set of Selected Factors: Cardiac In-Stent-Restenosis Predictors

	Name	Type	Values	
F1	Stent Type	Categorical	Bare Metal Stent (BMS)=1	Drug Eluting Stent (DES)=0
F2	Calcific Lesion	0/1 Categorical	Existence=1	Absence=0
F3	Prior PTCA	0/1 Categorical	Existence=1	Absence=0
F4	Prior MI	0/1 Categorical	Existence=1	Absence=0
F5	Stent Size	Continuous	[2mm, 4mm]	
F6	Chronic Renal Disease	0/1 Categorical	Existence=1	Absence=0
F7	Hyperlipidemia	0/1 Categorical	Existence=1	Absence=0
F8	Target Vessel	Categorical	Saphenous Vein Graft (SVG)=1	Native=0

The data used in this study is obtained from Ondokuz Mayıs University Hospital, Cardiology Department, based on the records of coronary stented cardiac patients of Prof. Dr. Mahmut Şahin, MD. A total of 10,435 records of cardiac patients between the years of 2005 and 2016 are scanned. Only the patients that are diagnosed with coronary heart disease, had coronary angiography operation, have at least one stented lesion and have at least six months of (mostly one year, maximum of three years) clinical and/or angiographical follow up period are eligible to be included in this study. The observations that satisfy all these conditions simultaneously in our dataset is 420. There are cases where a patient has more than one lesion that satisfy these conditions, however we use only one lesion from each patient to have independent observations. Therefore, number of observations decreases to 303. Among them, 63 lesions have in-stent-restenosis.

Before we apply our method, if a factor under consideration is continuous, observed patient value with respect to this factor is scaled to the interval [0,1] as follows in order to normalize its effect in the model:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}. \tag{4.1}$$

Since the categorical factors take values in interval [0,1], no action is needed. Additionally, to satisfy the monotonicity assumption of the model, the factor values are adjusted in such a way that higher values are more likely to have in-stent-restenosis.

4.3 Computational Analysis

In this section, we discuss the performances of the proposed algorithms on the instent-restenosis data mentioned in Section 4.2. To see the effect of integrating evolutionary algorithms with PCM, we begin by comparing the performances of PCM+NSGA-II and PCM+RECGA with Random+NSGA-II and Random+RECGA, respectively. The computational experiments of these models are conducted with NetBeans IDE 7.3 and CPLEX 12.6 on an Intel(R) Core(TM) i5-2410M 2.3GHz PC with 4 GB RAM, running under the Windows operating system.

Next, we compare the performances of PCM+NSGA-II and PCM+RECGA with some widely known machine learning methods: Logistic Regression (LR), Penalized Logistic Regression (pen-LR), Support Vector Machine (SVM), Artificial Neural Network (ANN), Decision Tree (DT) and Random Forest (RF). We conduct the experiments of LR and pen-LR on R and the other machine learning methods on MATLAB.

The classification threshold is determined as 0.5 in LR and pen-LR (i.e. they classify an observation as positive if the resulting probability of the algorithm is greater than 0.5). In SVM, the model trains itself using the radial basis kernel and utilizes an automatic hyper-parameter optimization, which selects the hyper-parameters that minimize 5-fold CV loss. There are two hidden layers in ANN.

Henceforth, these models are referred as *competitor models*, because we compare their performances with PCM+NSGA-II and PCM+RECGA.

We test the model performances by randomly splitting the data into mutually exclusive training set ($\mathcal{S}$), validation set ($\mathcal{V}$) and test set ($\widetilde{\mathcal{S}}$), and we repeat this procedure 100 times, where each instance is generated by a different random seed. Then, we report the average performances of these 100 instances and their standard deviations. For comparability, the experimental analysis of each model is conducted with the same 100 instances.

Note that, PCM+NSGA-II and PCM+RECGA first utilize $\mathcal{S}$ to generate initial set of solutions using PCM, and then they tune these solutions with $\mathcal{V}$ via evolutionary algo-

rithms. On the other hand, since Random+NSGA-II and Random+RECGA generate the initial solutions randomly, they just use $\mathcal{V}$ for training. Competitor models utilize $\mathcal{S} \cup \mathcal{V}$ as the training set. All the models' performances are tested in $\widetilde{\mathcal{S}}$.

Sensitivity, specificity and accuracy are identified as performance indicators and Fscore is reported as an indicator of balance between sensitivity and specificity. Besides them, Fmeasure is also reported.

We created two different settings, which are differentiated by the ratio of the number of patients with and without restenosis in training, validation and test samples. The details of the settings are given in Table 4.4.

Table 4.4: In-Stent-Restenosis Dataset, Settings

	Setting 1			**Setting 2**		
	$\mathcal{S}$	$\mathcal{V}$	$\widetilde{\mathcal{S}}$	$\mathcal{S}$	$\mathcal{V}$	$\widetilde{\mathcal{S}}$
# of patients with restenosis	24	24	12	24	24	12
# of patients without restenosis	96	96	48	24	24	12
Total # of patients	120	120	60	48	48	24
L for PCM	$\{0, \ldots, 96\}$			$\{0, \ldots, 24\}$		

For the instances for which Setting 1 is applied, there are 24 positive and 96 negative observations in training ($\mathcal{S}$) and validation ($\mathcal{V}$) samples, where the number of positive and negative observations in test sample ($\widetilde{\mathcal{S}}$) are 12 and 48, respectively. On the other hand, for each instance created as in Setting 2, there are 24 positive and 24 negative observations in training ($\mathcal{S}$) and validation ($\mathcal{V}$) sets, and the test set ($\widetilde{\mathcal{S}}$) is comprised of 12 positive and 12 negative observations.

Recall that, PCM is trained with $\mathcal{S}$ and it is solved for different values of parameter L. Since L represents the number of false positive observations that the model allows, it takes values between 0 and number of negative observations in sample $\mathcal{S}$. The last row of Table 4.4 indicates the values that the parameter L of PCM takes for the given settings.

Model hyper-parameters are set according to previously explained hyper-parameter

optimization process and they are given in Table 4.5. A detailed explanation of hyper-parameter tuning process, conducted with in-stent-restenosis dataset, can be found in the Appendix (Section C).

Table 4.5: Optimal Values of Hyper-parameters with Respect to In-Stent-Restenosis Dataset

		Setting 1	**Setting 2**
NSGA-II	*PopulationSize*	1000	1000
	GenerationSize	50	100
	NumberOfGenerations	5	100
	p_{rc}, p_{lc}	0.5, 0.5	0.5, 0.5
	p_m	0.01	0.01
RECGA	*PopulationSize*	250	150
	minFinalSetSize	100	50
	p_{rc}, p_{lc}	1.0, 0.0	0.5, 0.5
	p_m	0.5	0.01

4.4 Results

4.4.1 Role of PCM to Generate Initial Solutions to the Evolutionary Algorithms

In order to evaluate the effect of generating initial solutions of the evolutionary algorithms via PCM or random, Tables 4.6 and 4.7 give the training and test performances of PCM+NSGA-II, Random+NSGA-II, PCM+RECGA and Random+RECGA.

The tables indicate that the performances of Random+NSGA-II and Random+RECGA are biased towards one of the classes and they have highly unbalanced classification results. High specificity and poor sensitivity rates of these models imply that, Random+NSGA-II and Random+RECGA tend to classify most of the patients as negative, for this dataset. Similar results are observed for training and test performances for both settings.

Table 4.6: Training Performances: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA

	PCM+NSGA-II	Random+NSGA-II	PCM+RECGA	Random+RECGA
AVERAGE PERFORMANCE RESULTS				
	Setting1			
Sensitivity	0.64	0.26	0.75	0.46
Specificity	0.81	0.92	0.71	0.84
Accuracy	0.78	0.79	0.71	0.76
Fscore	0.71	0.40	0.72	0.58
Fmeasure	0.55	0.33	0.51	0.43
	Setting2			
Sensitivity	0.66	0.46	0.77	0.37
Specificity	0.70	0.79	0.69	0.87
Accuracy	0.68	0.62	0.73	0.62
Fscore	0.67	0.57	0.72	0.48
Fmeasure	0.67	0.54	0.74	0.47
STANDARD DEVIATIONS OF PERFORMANCE INDICATORS				
	Setting1			
Sensitivity	0.10	0.12	0.08	0.12
Specificity	0.10	0.10	0.07	0.08
Accuracy	0.09	0.08	0.05	0.05
Fscore	0.08	0.15	0.04	0.10
Fmeasure	0.08	0.13	0.05	0.08
	Setting2			
Sensitivity	0.16	0.15	0.09	0.20
Specificity	0.17	0.18	0.09	0.11
Accuracy	0.15	0.14	0.06	0.08
Fscore	0.15	0.15	0.06	0.19
Fmeasure	0.15	0.15	0.06	0.19

Table 4.7: Test Performances: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA

	PCM+NSGA-II	Random+NSGA-II	PCM+RECGA	Random+RECGA
AVERAGE PERFORMANCE RESULTS				
	Setting1			
Sensitivity	0.58	0.24	0.71	0.45
Specificity	0.79	0.91	0.68	0.82
Accuracy	0.75	0.78	0.69	0.74
Fscore	0.65	0.36	0.68	0.55
Fmeasure	0.48	0.29	0.47	0.40
	Setting2			
Sensitivity	0.63	0.44	0.72	0.39
Specificity	0.66	0.76	0.61	0.85
Accuracy	0.65	0.60	0.66	0.62
Fscore	0.62	0.53	0.64	0.49
Fmeasure	0.63	0.51	0.68	0.47
STANDARD DEVIATIONS OF PERFORMANCE INDICATORS				
	Setting1			
Sensitivity	0.17	0.15	0.15	0.18
Specificity	0.11	0.10	0.09	0.10
Accuracy	0.09	0.09	0.07	0.07
Fscore	0.13	0.18	0.08	0.16
Fmeasure	0.11	0.14	0.09	0.13
	Setting2			
Sensitivity	0.20	0.19	0.15	0.21
Specificity	0.20	0.21	0.16	0.14
Accuracy	0.16	0.15	0.10	0.08
Fscore	0.16	0.18	0.11	0.20
Fmeasure	0.17	0.18	0.10	0.20

Table 4.8 gives solution times of PCM+NSGA-II, Random+NSGA-II, PCM+RECGA and Random+RECGA.

Table 4.8: Solution Times (in sec.)

	PCM+NSGA-II	Random+NSGA-II	PCM+RECGA	Random+RECGA
	Setting 1			
AVG.	14.3	6.75	8.25	0.83
STD.DEV.	1.79	0.16	1.86	0.31
	Setting 2			
AVG.	103.65	104.3	0.86	0.23
STD.DEV.	3.02	3.71	0.35	0.06

4.4.2 Comparison of PCM+NSGA-II, PCM+RECGA and Competitor Models

In this subsection, we compare the performances of PCM+NSGA-II and PCM+RECGA with those of competitor models. Their training and test performances are given in Tables 4.9 and 4.10, respectively. The training performance of a model evaluates its fit to the dataset given to tune the model parameters. Then, to see the performance of a model on a dataset which is comprised of previously unseen observations, we test it on a test set. The models' performance to react to new data represents is generalization ability. Note that, a high training and low test performance indicate that the model overfits to the training set and its generalization ability is poor.

The model performances are investigated under two different settings, as explained previously. The ratio of positive observations to all observations are 20% and 50%, in Setting 1 and Setting 2, respectively. The number of positive observations do not change and number of negative observations used in Setting 1 is higher than that of Setting 2.

Table 4.9: Training Performances

	PCM+NSGA-II	PCM+RECGA	LR	pen-LR	SVM	ANN	DT	RF
AVERAGE PERFORMANCE RESULTS								
Setting 1								
Sensitivity	0.64	0.75	0.31	0.31	0.16	0.35	0.43	0.56
Specificity	0.81	0.71	0.98	0.98	0.99	0.98	0.97	0.97
Accuracy	0.78	0.71	0.85	0.85	0.83	0.86	0.86	0.88
Fscore	0.71	0.72	0.47	0.47	0.23	0.51	0.59	0.71
Fmeasure	0.55	0.51	0.45	0.45	0.23	0.49	0.54	0.66
Setting 2								
Sensitivity	0.66	0.77	0.75	0.75	0.85	0.83	0.82	0.90
Specificity	0.70	0.69	0.73	0.74	0.73	0.75	0.80	0.84
Accuracy	0.68	0.73	0.74	0.74	0.79	0.79	0.81	0.87
Fscore	0.67	0.72	0.74	0.74	0.78	0.78	0.81	0.86
Fmeasure	0.67	0.74	0.74	0.75	0.80	0.79	0.81	0.87
STANDARD DEVIATIONS OF PERFORMANCE INDICATORS								
Setting 1								
Sensitivity	0.10	0.08	0.04	0.04	0.17	0.10	0.07	0.07
Specificity	0.10	0.07	0.01	0.01	0.01	0.02	0.01	0.01
Accuracy	0.09	0.05	0.01	0.01	0.03	0.01	0.01	0.01
Fscore	0.08	0.04	0.05	0.05	0.25	0.09	0.07	0.05
Fmeasure	0.08	0.05	0.04	0.04	0.24	0.07	0.05	0.04
Setting 2								
Sensitivity	0.16	0.09	0.05	0.05	0.06	0.11	0.05	0.05
Specificity	0.17	0.09	0.06	0.06	0.11	0.10	0.05	0.05
Accuracy	0.15	0.06	0.04	0.04	0.07	0.04	0.02	0.03
Fscore	0.15	0.06	0.04	0.04	0.08	0.06	0.03	0.03
Fmeasure	0.15	0.06	0.04	0.03	0.06	0.06	0.03	0.02

Table 4.10: Test Performances

	PCM+NSGA-II	PCM+RECGA	LR	pen-LR	SVM	ANN	DT	RF
AVERAGE PERFORMANCE RESULTS								
Setting 1								
Sensitivity	0.58	0.71	0.32	0.31	0.11	0.27	0.32	0.34
Specificity	0.79	0.68	0.97	0.97	0.98	0.95	0.93	0.91
Accuracy	0.75	0.69	0.84	0.84	0.81	0.82	0.81	0.80
Fscore	0.65	0.68	0.46	0.45	0.16	0.40	0.45	0.47
Fmeasure	0.48	0.47	0.43	0.42	0.15	0.36	0.39	0.38
Setting 2								
Sensitivity	0.63	0.72	0.73	0.73	0.78	0.71	0.67	0.73
Specificity	0.66	0.61	0.66	0.67	0.59	0.65	0.68	0.66
Accuracy	0.65	0.66	0.69	0.70	0.68	0.68	0.68	0.69
Fscore	0.62	0.64	0.67	0.68	0.65	0.65	0.65	0.67
Fmeasure	0.63	0.68	0.70	0.70	0.71	0.68	0.67	0.70
STANDARD DEVIATIONS OF PERFORMANCE INDICATORS								
Setting 1								
Sensitivity	0.17	0.15	0.13	0.13	0.14	0.14	0.14	0.15
Specificity	0.11	0.09	0.03	0.03	0.03	0.05	0.04	0.04
Accuracy	0.09	0.07	0.03	0.03	0.02	0.04	0.04	0.04
Fscore	0.13	0.08	0.15	0.15	0.20	0.16	0.16	0.17
Fmeasure	0.11	0.09	0.14	0.14	0.19	0.14	0.14	0.14
Setting 2								
Sensitivity	0.20	0.15	0.14	0.14	0.13	0.17	0.16	0.15
Specificity	0.20	0.16	0.15	0.15	0.16	0.18	0.15	0.16
Accuracy	0.16	0.10	0.09	0.09	0.09	0.10	0.10	0.09
Fscore	0.16	0.11	0.11	0.10	0.11	0.13	0.11	0.11
Fmeasure	0.17	0.10	0.10	0.10	0.09	0.12	0.11	0.10

According to the performances given in Tables 4.9 and 4.10, Figure 4.1 illustrates the training and test performances of the models together.

It is observed that, for Setting 1, the competitor models' sensitivity values are considerably low and specificity values are extremely high, in both training and test. This implies that, the classification ability of these models, under Setting 1, is poor. In terms of Fscore, training performances of PCM+NSGA-II (0.71), PCM+RECGA (0.72) and RF (0.71) are the highest. For the test performances, the best Fscore of Setting 1 belongs to PCM+RECGA (0.68), followed by PCM+NSGA-II (0.65). However, despite its high training performance, RF's Fscore in test is weak (0.47).

In Setting 2, the performances of the competitor models are more promising than Setting 1, yet, the performances of PCM+NSGA-II and PCM+RECGA do not change significantly. Thus, we can say that, our models are more robust than the competitor models, in spite of the changes in number of positive and negative observations in training sample. In terms of Fscore, highest training performances of Setting 2 belong to RF, DT, SVM, ANN, LR, pen-LR, PCM+RECGA and PCM+NSGA-II, respectively. For test performances, it is observed that, the Fscore values of all models are close to each other and ranging between 0.62 to 0.68.

As previously mentioned, in Setting 1, where the ratio of positive observations is 20%, the sensitivity values of competitor models in training are extremely low while they have high specificity values. This may indicate that, their training phase terminates when a certain level of training accuracy is satisfied. Since negative observations constitutes the majority class in the training sample, classifying most of the observations as negative yields high accuracy values even the sensitivity is low. In Setting 2, despite the number of observations in training set is fewer than Setting 1, there are equal number of positive and negative observations. Thus, to keep the training accuracy high, the competitor models have to obtain high classification results for both classes. This may be the reason why the competitor models have high Fscore performances in Setting 2.

It is also observed that, for Setting 1, for both of training and test, the specificity performance of each model is higher than Setting 2. In a similar manner, for Setting 2, the sensitivity performances are higher than that of Setting 1. In a medical diagnosis

problem, the ideal case is to have extremely high values of sensitivity and specificity, simultaneously. If this is not possible, lower specificity for the sake of high sensitivity is the second best option. This is because, while a false negative classification may result in serious health problems, a false positive classification causes just financial burden.

Considering all these arguments, it can be said that, if it is possible, having a training sample as large as possible which contains balanced amount of observations from each class may result in high classification performances. However, if the number of positive observations is few but there are many negative observations, by keeping the number of positive and negative observations equal in training sets, a model can provide higher sensitivity rates.

Note that, since Fmeasure is not one of our performance indicators, we do not give a detailed analysis about it. Yet still it can be said that, our model performances can compete with those of the competitor models.

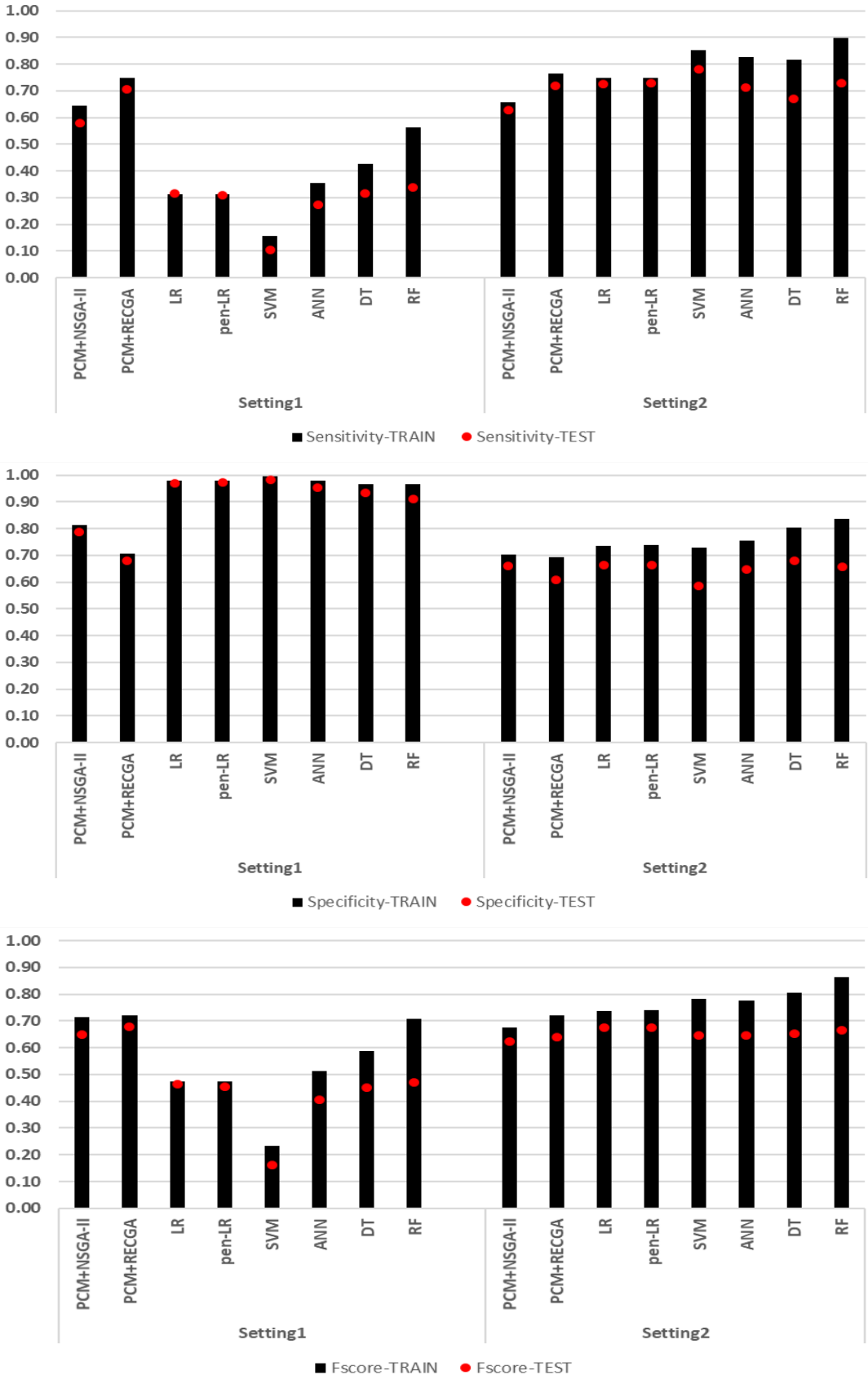


Figure 4.1: Training vs. Test Performances

Figure 4.2 illustrates the gaps between the models' training and test performances. The yellow line in the graphics indicates the average gap of the models. Note that, the gap refers to how much the training performance is greater than the test performance and if the training performance lags behind the test, the gap is expressed as zero.

It is observed that, for both settings, the models whose training and test gaps are always below the average gap are PCM+NSGA-II, PCM+RECGA, LR and pen-LR. High gaps between training and test indicate that, even if a model has a good performance in training, its test performance is low. Thus, the model overfits to the training set and the generalization ability (which refers the ability of the model in adapting to new observations represented by the test set) of such a model is poor because it does not perform well to predict the classes of unseen observations. Although the compatibility of training and test performances of a model reflects its level of generalizability, having high performances in training and test is a must for good generalization. However, the performances of LR and pen-LR, in Setting 1, are quite poor.

We can say that, PCM+NSGA-II and PCM+RECGA are reliable and successful models both because of their robustness against the configuration of samples and their generalization ability. All competitor models solve an instance within a minute, mostly in seconds. Thus, we do not give their specific solution times.

When the performances of PCM+NSGA-II and PCM+RECGA are compared, it can be observed that, even though the Fscore values of PCM+NSGA-II and PCM+RECGA are similar, PCM+RECGA has higher sensitivity and lower specificity compared to PCM+NSGA-II. Additionally, PCM+RECGA has lower standard deviations, and more robust against the changes in number of observations in training, validation and test samples.

Detailed tables that give the performances of the models, PCM+NSGA-II and PCM+RECGA in $\mathcal{S}$, $\mathcal{V}$ and $\widetilde{\mathcal{S}}$; and Random+NSGA-II and Random+RECGA in $\mathcal{S}$, $\mathcal{V}$ and $\widetilde{\mathcal{S}}$, separately, can be found in the Appendix (Section I). Besides the ratios of correct classifications, number of true positive, true negative and total true classifications are also given in these tables.

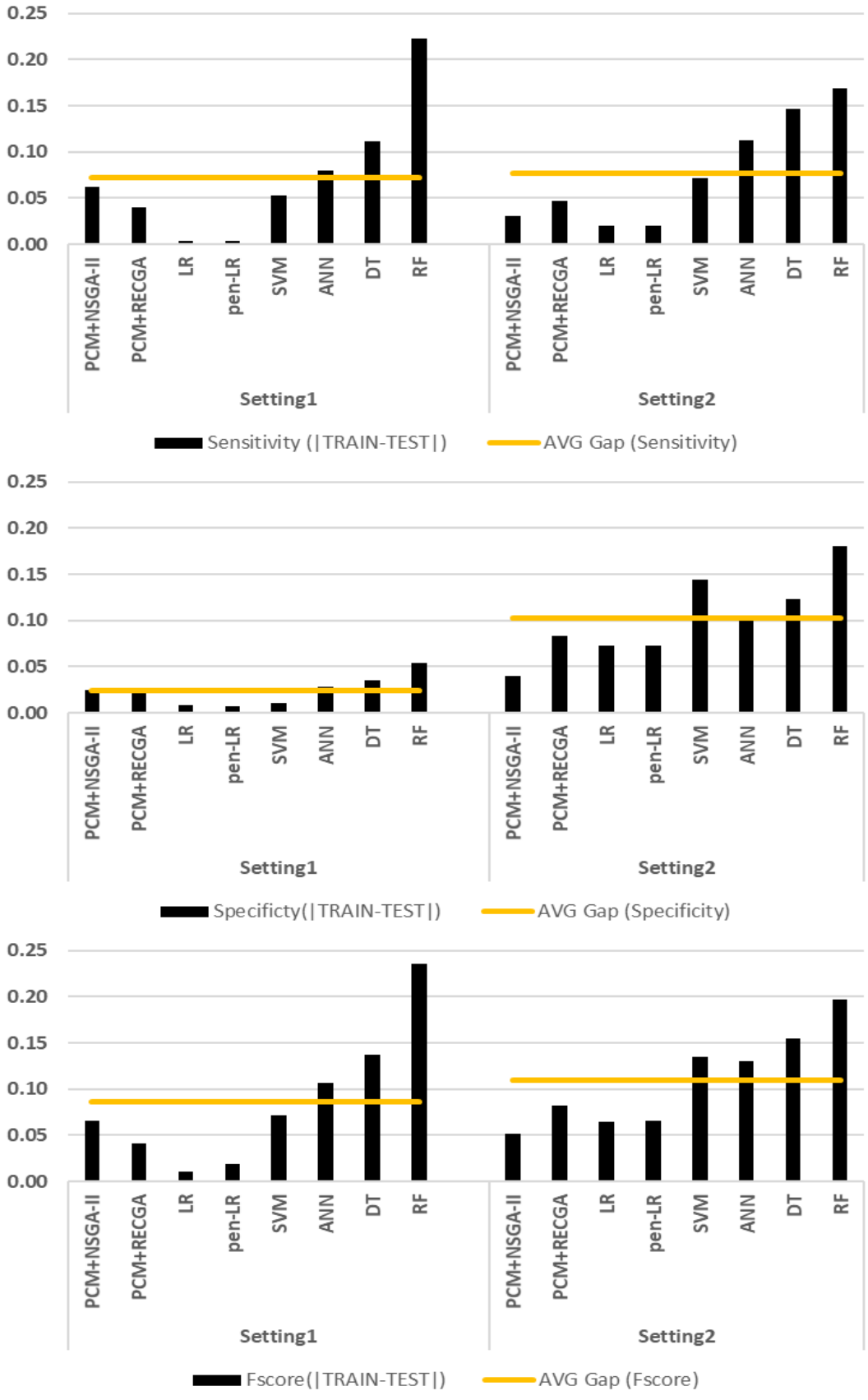


Figure 4.2: Gap Between Training and Test Performances

4.5 Prediction Performances of Cardiologists vs. PCM+NSGA-II and PCM+RECGA

In this section, to evaluate the efficiency of PCM+NSGA-II and PCM+RECGA, we report the experimental analysis where we compare the classification performance of these models and a group of doctors who are specialized in the area of coronary in-stent-restenosis.

It is crucial to have high sensitivity in the case where the diagnosis test is used to detect a serious but treatable disease. In the cases where the sensitivity of the test is high but specificity is low, there will be patients who do not actually have the disease but undergone further investigation, due to false positive results [120].

In this classification problem, our model prioritizes correct classification of the patients who has restenosis in reality. It also gives particular importance to having a true negative rate as high as possible.

As it is discussed in Section 4.3, having a training set as large as possible which contains balanced amount of positive and negative observations may result in high classification performances. Therefore, the training sample of this study is created as the most crowded training set in which the observations in the two classes are equal.

Recall that, the dataset is comprised of 63 patients with in-stent restenosis, and 240 patients without in-stent-restenosis. Therefore, 124 patients are randomly selected in order to construct the samples $\mathcal{S}$ and $\mathcal{V}$, such that each sample has equal number of patients from each side and equal number of observations are included in both samples.

In this setting, we run PCM+NSGA-II and PCM+RECGA to classify a single patient in the test set and we repeat this procedure for 100 times. In half of the experiments, test set, $\widetilde{\mathcal{S}}$, comprised of a positive observation. Therefore, models make predictions for 50 patients with restenosis and 50 patients without restenosis. At each run, models randomly construct mutually exclusive $\mathcal{S}$, $\mathcal{V}$ and $\widetilde{\mathcal{S}}$. Thus, the training ($\mathcal{S}$), validation ($\mathcal{V}$) and test ($\widetilde{\mathcal{S}}$) samples of this study are created as it is shown in Table 4.11.

Since, the sample configurations of the experiment resembles Setting 2, the hyper-

parameter values of the models are set, accordingly.

Table 4.11: Sample Configurations of the Experiment

	Dataset	$\mathcal{S}$	$\mathcal{V}$	$\widetilde{\mathcal{S}}$		
Total	**303**	**62**	**62**	**1**		
# patients with restenosis	63	31	31	1	OR	0
# patients without restenosis	240	31	31	0		1

15 cardiologists have participated in this study. They are employed in cardiology departments of universities, public or private hospitals. We assume that their experience in the field of medicine is proportional to the years since their graduation from medical school. Similarly, their experience in the field of cardiology is defined with the years since they get their graduate degree in cardiology. In this context, the experience of the doctors who have joined our study ranges between 9 to 35 years, with an average of 17.8 years, while their experience in the field of cardiology ranges between 3 to 27 years, with an average of 11.47 years.

Note that, medical experiences of the cardiologists serve like a training set. Considering the given years of experiences, it is expected that each medical doctor joined in this study have seen more patients with and without restenosis compared to number of observations used to train the models (which consists of 124 patients, 62 with and 62 without in-stent-restenosis, i.e. $|\mathcal{S} \cup \mathcal{V}|$).

We have provided the data of the 100 test patients to the 15 cardiologists and asked them to make predictions about their restenosis status by using the given values of predictors as given in Table 4.3. The predictor values and actual situations of these 100 patients can be found in Table J.1 in the Appendix (Section J).

Table 4.12 summarizes the prediction performances of medical doctors, PCM+NSGA-II and PCM+RECGA.

Table 4.12: Prediction Performances of Medical Doctors, PCM+NSGA-II and PCM+RECGA

	Sensitivity	**Specificity**	**Accuracy**	**PPV**	**NPV**	**FPR**	**FNR**	**Fscore**	**Fmeasure**
MD1	0.42	0.86	0.64	0.75	0.60	0.14	0.58	0.56	0.54
MD2	0.32	0.92	0.62	0.80	0.58	0.08	0.68	0.47	0.46
MD3	0.22	0.94	0.58	0.79	0.55	0.06	0.78	0.36	0.34
MD4	0.22	0.96	0.59	0.85	0.55	0.04	0.78	0.36	0.35
MD5	0.08	1.00	0.54	1.00	0.52	0.00	0.92	0.15	0.15
MD6	0.34	0.98	0.66	0.94	0.60	0.02	0.66	0.50	0.50
MD7	0.72	0.44	0.58	0.56	0.61	0.56	0.28	0.55	0.63
MD8	0.40	0.96	0.68	0.91	0.62	0.04	0.60	0.56	0.56
MD9	0.48	0.70	0.59	0.62	0.57	0.30	0.52	0.57	0.54
MD10	0.44	0.96	0.70	0.92	0.63	0.04	0.56	0.60	0.59
MD11	0.30	1.00	0.65	1.00	0.59	0.00	0.70	0.46	0.46
MD12	0.64	0.82	0.73	0.78	0.69	0.18	0.36	0.72	0.70
MD13	0.48	0.70	0.59	0.62	0.57	0.30	0.52	0.57	0.54
MD14	0.44	0.88	0.66	0.79	0.61	0.12	0.56	0.59	0.56
MD15	0.64	0.80	0.72	0.76	0.69	0.20	0.36	0.71	0.70
AVG. of MD	**0.41**	**0.86**	**0.64**	**0.80**	**0.60**	**0.14**	**0.59**	**0.52**	**0.51**
STD.DEV. of MD	0.17	0.15	0.05	0.13	0.05	0.15	0.17	0.14	0.14
PCM+NSGA-II	**0.64**	**0.78**	**0.71**	**0.74**	**0.68**	**0.22**	**0.36**	**0.70**	**0.69**
PCM+RECGA	**0.70**	**0.68**	**0.69**	**0.69**	**0.69**	**0.32**	**0.30**	**0.69**	**0.69**

Figure 4.3 graphs the sensitivity, specificity, accuracy and Fscore values for each medical doctor, PCM+NSGA-II and PCM+RECGA. It clearly shows the robustness of PCM+NSGA-II and PCM+RECGA in classification of patients.

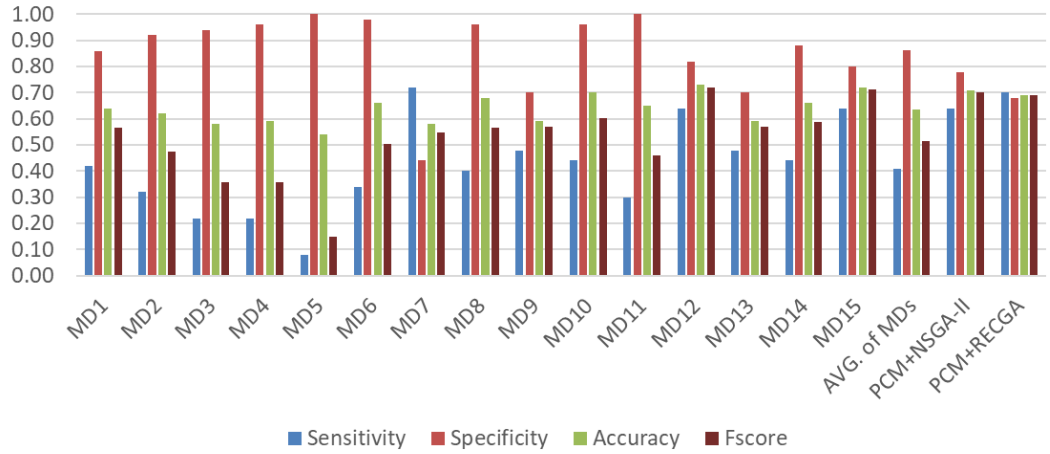


Figure 4.3: Prediction Performances - Medical Doctors vs. PCM+NSGA-II and PCM+RECGA

Results (Table 4.12 and Figure 4.3) indicate that sensitivity of medical doctors ranges between 0.08 and 0.72 with an average value of 0.41. Specificity of medical doctors ranges between 0.44 and 1.00 with the average of 0.86. Sensitivity and specificity of PCM+NSGA-II and PCM+RECGA are 0.64, 0.78, 0.70 and 0.68, respectively.

Overall prediction accuracy of medical doctors takes values between 0.54 and 0.73 with an average of 0.64. Prediction accuracy of PCM+NSGA-II and PCM+RECGA are 0.71 and 0.69, respectively. The performance of PCM+NSGA-II is greater than 13 out of 15 medical doctors' prediction accuracy. The accuracy results of the other two cardiologists are 0.72 and 0.73. The accuracy performance of PCM+RECGA is greater than 12 out of 15 medical doctors' prediction accuracy, where the accuracy results of the other three cardiologists are 0.70, 0.72 and 0.73.

It is important to note that since we need high values of sensitivity and specificity simultaneously, the performance indicator Fscore, combining the sensitivity and specificity into a single measure, is important for us. In terms of Fscore values, the score of medical doctors ranges between 0.15 and 0.72 with an average of 0.52. On the other hand, Fscore of PCM+NSGA-II and PCM+RECGA are 0.70 and 0.69, respectively.

Models' performances are better than 13 out of 15 medical doctors' scores.

It is observed that, medical doctors have high specificity values in general. However, for the ones whose specificity values are high, the maximum value of sensitivity is 0.64. Highest sensitivity achieved by a medical doctor is 0.72, but in this case, the specificity value is only 0.44. Also, note that, for only two doctors (MD12 and MD15) both sensitivity and specificity are above 0.5 (i.e. better than a random classifier).

When we evaluate the performance of PCM+NSGA-II in terms of the same performance indicators, we observe that, compared to most of the medical doctors, model achieves relatively balanced sensitivity and specificity rates. Its Fscore value is higher than 13 out of all the medical doctors'.

PCM+RECGA achieves high sensitivity and specificity rates, too. The performances of PCM+RECGA is much more balanced compared to the performances of PCM+NSGA-II. Its Fscore value is also higher than 13 out of all of the medical doctors'.

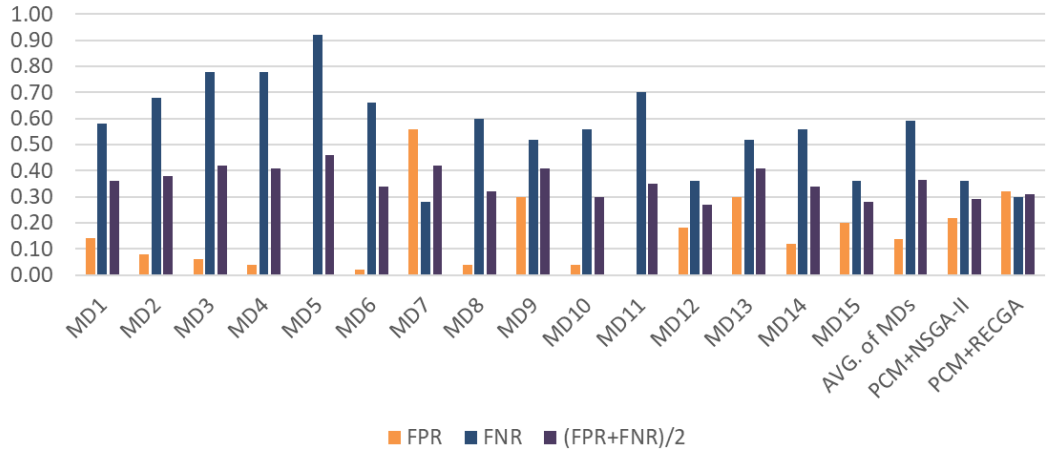


Figure 4.4: False Prediction Performances - Medical Doctors vs. PCM+NSGA-II and PCM+RECGA

Figure 4.4 gives the false positive and false negative ratios together with their average. False positive ratio of medical doctors ranges between 0 and 0.56 with an average of 0.14. False negative ratio of medical doctors takes values between 0.28 and 0.92 with an average of 0.59. It is worth to note that, low FPR and high FNR of medical

doctors are indications of their small number of positive and large number of negative classifications. FPR and FNR of PCM+NSGA-II are 0.22 and 0.36, respectively. For PCM+RECGA, FPR is 0.32 and FNR is 0.30.

We combine the FPR and FNR into a single measure by taking average of these two values. It is desired to have simultaneously low values of FPR and FNR. Combined measure of medical doctors takes values between 0.27 and 0.46, with the average value of 0.36, while the same measure of PCM+NSGA-II and PCM+RECGA are 0.29 and 0.31, respectively. The performance of PCM+NSGA-II is better than 13 out of the 15 medical doctors' values. The value of this measure for the remaining doctors are 0.27 and 0.28. For PCM+RECGA, the same performance indicator is better than 12 out of the 15 medical doctors' values. For the remaining three doctors, this measure takes values of 0.27, 0.28 and 0.30.

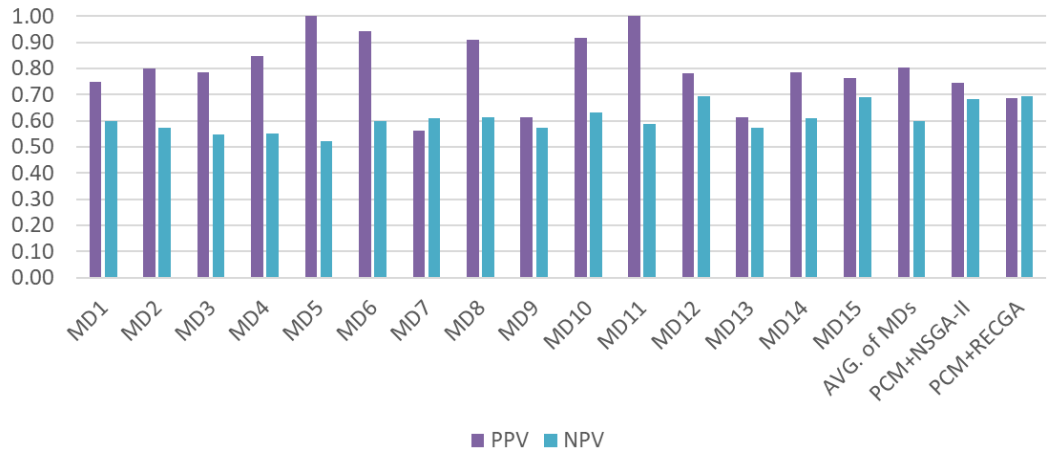


Figure 4.5: Positive and Negative Predictive Values - Medical Doctors vs. PCM+NSGA-II and PCM+RECGA

Figure 4.5 gives the graphs of the positive and negative predictive values. PPV of medical doctors takes values between 0.56 and 1 with an average value of 0.80. NPV of medical doctors ranges between 0.52 and 0.69 with an average of 0.60. PPV and NPV of PCM+NSGA-II are 0.74 and 0.68, respectively. This indicates that, among the patients whom PCM+NSGA-II classified as positive, 74% correctly have the restenosis, and among the patients whom PCM+NSGA-II classified as negative, 68% correctly do not have the disease. For PCM+RECGA, PPV and NPV are both

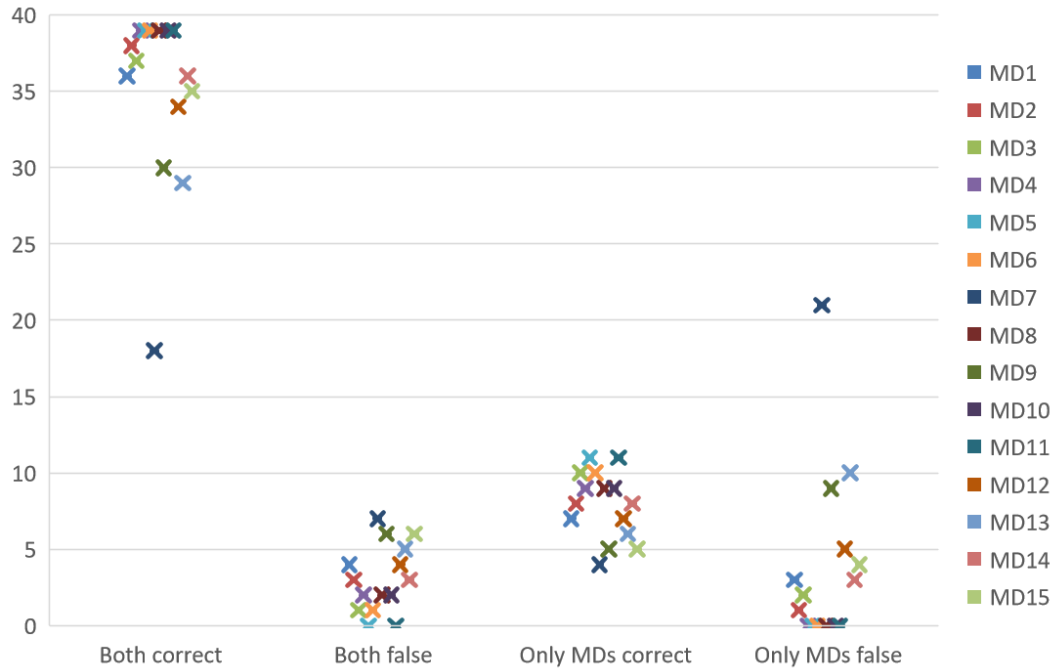


Figure 4.6: Classification Numbers: PCM+NSGA-II vs. Medical Doctors - Patients Without Restenosis

0.69.

Figures 4.6 and 4.7 (Figures 4.8 and 4.9) illustrate the prediction performances of medical doctors and PCM+NSGA-II (PCM+RECGA), respectively. We can divide the predictions into four groups based on the decisions of the doctors and the model. First and second group of Figures 4.6, 4.7, 4.8 and 4.9 represent the number of patients that are classified correctly and incorrectly both by the medical doctors and the model, respectively. Third group represents the number of patients that are classified correctly by the medical doctors but incorrectly by the model, and fourth group stands for the number of patients that are classified correctly by the model but incorrectly by the medical doctors. Each indicator under a group corresponds to a medical doctor. For example, the data represented by the leftmost sign of first group of Figure 4.6 represent that there are 36 patients classified correctly both by MD1 and PCM+NSGA-II.

For the patients whose real restenosis status are negative, number of patients who are incorrectly classified by PCM+NSGA-II but correctly classified by the medical doctors takes values between 4 and 11 with an average of 7.9, and number of pa-

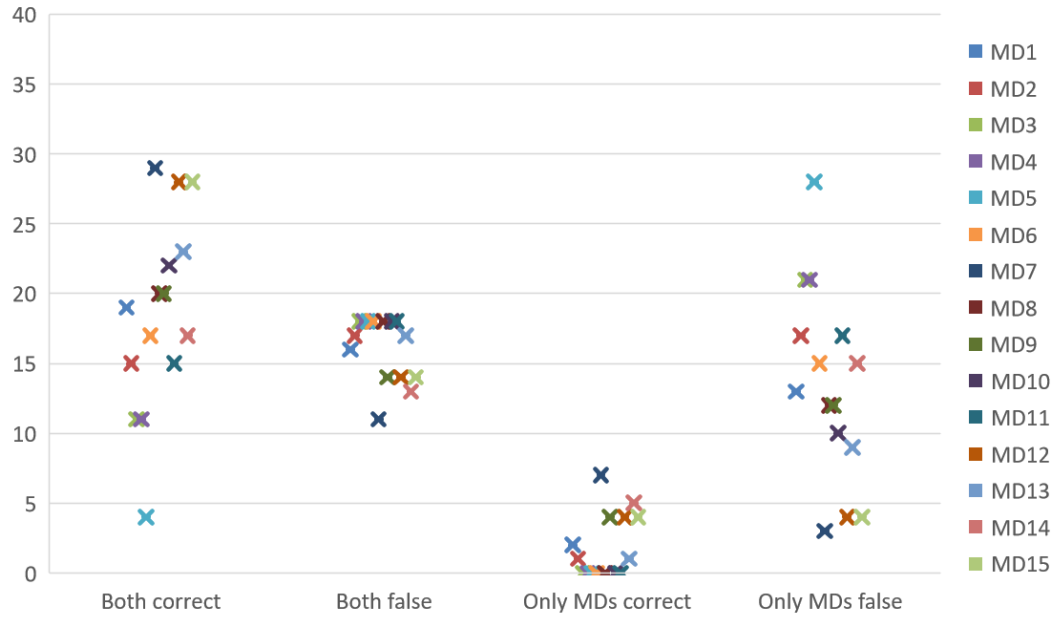


Figure 4.7: Classification Numbers: PCM+NSGA-II vs. Medical Doctors - Patients with Restenosis

tients who are incorrectly classified by the medical doctors but correctly classified by PCM+NSGA-II ranges between 0 and 21 with an average of 3.87.

For the patients with a positive restenosis status in reality, number of patients who are incorrectly classified by PCM+NSGA-II but correctly classified by the medical doctors takes values between 0 and 7 with an average of 1.87, and number of patients who are incorrectly classified by the medical doctors but correctly classified by PCM+NSGA-II takes values between 3 and 28 with an average of 13.4.

For the patients whose real restenosis status are negative, number of patients who are incorrectly classified by PCM+RECGA but correctly classified by the medical doctors takes values between 5 and 16 with an average of 12, and number of patients who are incorrectly classified by the medical doctors but correctly classified by PCM+RECGA ranges between 0 and 17 with an average of 3.

For the patients with a positive restenosis status in reality, number of patients who are incorrectly classified by PCM+RECGA but correctly classified by the medical doctors takes values between 0 and 7 with an average of 2.33, and number of patients who are incorrectly classified by the medical doctors but correctly classified by PCM+RECGA

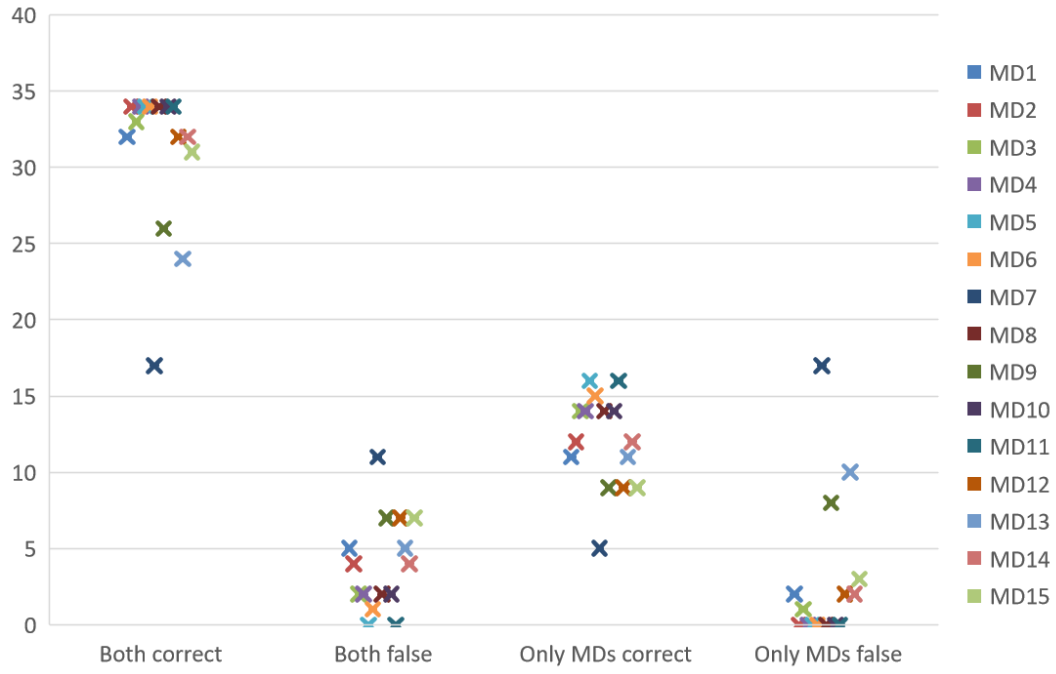


Figure 4.8: Classification Numbers: PCM+RECGA vs. Medical Doctors - Patients Without Restenosis

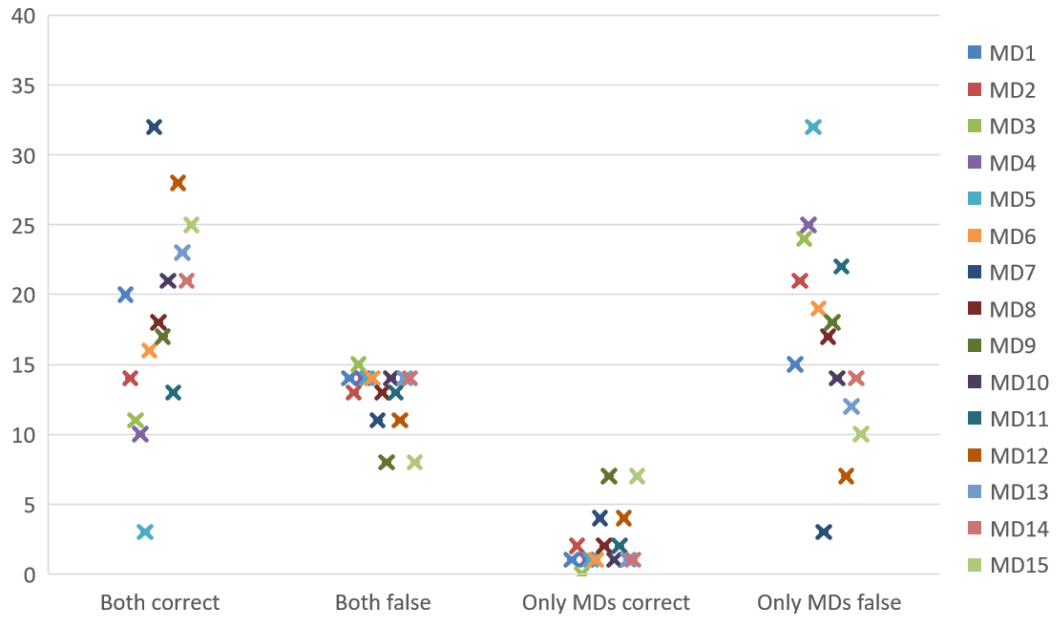


Figure 4.9: Classification Numbers: PCM+RECGA vs. Medical Doctors - Patients with Restenosis

takes values between 3 and 32 with an average of 16.87.

This indicates that, the number of patients who carries the disease and disregarded by the models, but correctly identified by the doctors, is very low. However, the number of patients in the opposite situation is quite high. This shows the strength of PCM+NSGA-II and PCM+RECGA against the medical specialists in detecting patients with the disease.

When PCM+NSGA-II and PCM+RECGA are compared, it is observed that, PCM+NSGA-II has higher specificity but lower sensitivity.

Note that, even though it can achieve better performances than most of the medical doctors, PCM+NSGA-II is dominated by MD12 and MD15. However, none of the medical doctors can dominate the performances of PCM+RECGA.

4.6 Conclusion

Since PCM+NSGA-II and PCM+RECGA are suggested as medical diagnostic models, our aim is to demonstrate that they are effective and reliable decision support tools for classification of patients. The experimental results indicate that, sensitivity, specificity and accuracy rates achieved by PCM+NSGA-II and PCM+RECGA are quite promising compared to the clinical detection methodologies, when their performance measures, risks and costs are considered together. Also they provide a great advantage by foreseeing the risk of restenosis at the time of stent implantation. In clinical situations, cardiologists recommend a course of action to a patient to confirm or deny the existence of the disease. Therefore, we can position our models as medical decision aids which assist experts in recommending the course of action by classifying patients according to their risk of the disease.

We have also compared the model performances with the classifications made by experts who utilized only the given values of in-stent-restenosis predictors and their personal experiences. As it is indicated before, the main objective of this problem is predicting the correct class of restenosis of patients in test set and it is important to achieve this with both high and balanced prediction results for the patients

with and without the disease. In this context, the comparison results of cardiologists, PCM+NSGA-II and PCM+RECGA indicate that, the proposed models achieve the above mentioned objectives and surpass the prediction abilities of majority of the medical doctors. The analysis of the performance indicators as a whole suggests that, even when half of the patients in test set have restenosis, the medical doctors have classified relatively low number of patients as positive and they have classified most of them as negative. This leads to an imbalanced performance in terms of true positive and true negative classifications. Thus, it is observed that, in general the medical doctors achieve higher prediction performances in specificity, however their sensitivity rates are quite low. The strongest feature of PCM+NSGA-II and PCM+RECGA is their balanced and simultaneously high classification performances of positive and negative observations. It is observed that, considering Fscore, PCM+NSGA-II and PCM+RECGA outperform almost all the cardiologists participated in this study. Just two cardiologists perform slightly better performances than the proposed models.

The comparison results of PCM+NSGA-II, PCM+RECGA and the competitor models suggest that, the proposed models are robust against the variations in samples used for training. Additionally, since their training and test performances are compatible with each other and promising, they are reliable and also their generalization ability (performance in test sample) is higher.

Thus, by evaluating all these findings together, we can say that, PCM+NSGA-II and PCM+RECGA are reliable and effective tools that are used for classification purposes. Specific to this problem, it can be concluded that, the given algorithms are promising decision support tools for cardiologists in the process of determining potential restenosis status of a patient and recommending a course of action.

Note that, the study in Chapter 4 was conducted in accordance with the Declearation of Helsinki, and the protocol was approved by the Ethics Committee of Ondokuz Mayıs University at date 07/27/2017 with project identification code of OMÜ-KAEK 2017/272.

CHAPTER 5

RARE EVENT CLASSIFICATION MODELS FOR MEDICAL DIAGNOSIS PROBLEM APPLIED TO BREAST CANCER

Breast cancer is the most common cancer in women, worldwide [121]. It can be seen in both women and men, but its frequency in women is much more higher. Even though it is identified that hormonal, lifestyle and environmental factors may affect the risk of developing breast cancer, its cause may be a complex interaction of genetic and environmental factors [122]. Mammograms, breast ultrasound, breast MRI scans or some other recently developed imaging tests can be utilized as the diagnosis methodologies of breast cancer. However, the gold standard is biopsy to diagnose or reject the existence of the disease [123].

In the literature, there are well studied, structured, proper and large size datasets about breast cancer. Two of them are Wisconsin Breast Cancer Original dataset (WBCO) [5] and Wisconsin Breast Cancer Diagnostic dataset (WBCD) [6] which are available in UCI Machine Learning Repository. The datasets are comprised of dichotomous dependent variables, that identifies the status of a tumor (malignant or benign).

In this part of our study, we test the performances of the developed models on these datasets. For the cases where the incidence of a disease among the population is low, it is harder to identify the existence of the disease. In this context, to evaluate the model performances when one class of observations are rare compared to other, we create class imbalance between the class of malignant and benign tumors and we test our models to see their performances in classifying observations that belong to class of rarely found members. We prefer to perform this analysis on WBCO and WBCD datasets based on the fact that these datasets are well structured and contains substantial amount of observations.

Following two sections give a brief literature review about the machine learning applications which employs WBCO and WBCD datasets, respectively.

5.1 Machine Learning Applications with the Wisconsin Breast Cancer Original Dataset

In the literature, there are many studies that work with the WBCO dataset that can be obtained from the Original Wisconsin Breast Cancer Database maintained by Dr. William H. Wolberg from University of Wisconsin. It is a well structured data including the breast cancer predictors and the class of the tumors (malignant or benign). There exists 699 instances (458 benign, 241 malignant) and 9 predictors in the dataset. The predictors are defined as the cytological characteristics of breast fine-needle aspirates, and are valued on an integer scale between 1 and 10, where a higher value is closer to malignancy. Table 5.1 shows the predictors of the dataset [5].

Table 5.1: Predictors of the WBCO Dataset

1. Clump thickness
2. Uniformity of cell size
3. Uniformity of cell shape
4. Marginal adhesion
5. Single epithelial cell
6. Bare nuclei
7. Bland chromatin
8. Normal nuclei
9. Mitoses

Table 5.2 summarizes the studies that work with this dataset. None of these studies consider the case where positive observations are rare in the population. The rarity of positive observations in the original dataset is 35% (malign:benign = 241:458 = 1.0:1.9), and the rarity of positive observations in training set is above 30% for most of the studies.

Rareness level of positive observations to all observations in the training and test sets

are given in fifth and sixth columns of Table 5.2, respectively, for the studies that report this information. Rest of the studies in Table 5.2 use samples disregarding this ratio. They create training and test samples by randomly choosing observations from all observations due to the predetermined training-test partition.

It is observed that, many of the studies conducted experiments with various training-test partitions (e.g. 80-20%, 70-30%, 50-50%), feature selection is conducted in some studies and most of the studies use cross-validation in the experiments. Since the classification performance of a model can be effected by these factors, we only state the maximum accuracy that a classifier achieved, in Table 5.2. Although most of the studies interested in the classification accuracy, Table 5.2 includes sensitivity and specificity for the studies which report these values.

Setiano conduct one of the earliest studies and propose an algorithm to prune a standard three-layer feed forward neural network (NN). The experiments are conducted on a single configuration where training set consists of 229 benign and 121 malignant observations. The rest of the data is used as test sample [124].

Another study of Setiano also utilizes neural networks. First, the author propose a neural network with one hidden unit for attribute selection. Then, experiments for breast cancer diagnosis are conducted with hundred neural networks each with three hidden units and hundred neural networks each with five hidden units. The training set includes 119 malignant and 222 benign observations, and the rest of the data is reserved for the test sample [125].

Pena-Reyes and Sipper propose a binary diagnosis algorithm and provides a numeric value that represents the confidence level of the system about the response variable. The experiments are conducted with three different configurations. Training set in the first configuration contains all cases and test set is empty, second configuration contains 75% of the cases and test set has 25% and third configuration contains 50% of the cases and the test set has 50% [126].

Table 5.2: Various Machine Learning Applications to the WBCO Dataset

Source	Method	Classification Accuracy (%)	Training-Test Partition(%)	Malignant Ratio in Training Set	Malignant Ratio in Test Set	Sensitivity (%)	Specifictiy (%)
[124]	Pruned-NN	96.56	50-50	0.35	0.34	96.67	96.51
[127]	C4.5	94.74	10-fold CV				
[128]	RIAC	94.99	Leave one out CV				
[129]	Fisher LDA	96.8	10-fold CV				
[130]	SVM	97.2	5-fold CV				
[131]	NEFCLASS	95.06	10-fold CV				
[126]	Fuzzy-Genetic Algorithm	97.8	100-0			97.07	98.7
[125]	Neuro-rule 2a	98.24	50-50	0.35	0.35	99.1	97.75
[132]	Optimized-LVQ	96.7	10-fold CV				
	Big LVQ	96.8	10-fold CV				
	AIRS	97.2	10-fold CV				
[133]	LSA with Perceptron Algorithm	98.8	50-50/75-25	0.55/0.73	0.65/0.65		
[134]	Supervised Fuzzy Clustering	95.57	10-fold CV				
[135]	LS-SVM	97.08	80-20	0.35	0.34	97.87	97.77
		98.53	10-fold CV				
[136]	SVM	99.54	37-63	0.32	0.37	99.37	99.64
	RNN	98.61	37-63	0.32	0.37	98.11	98.91
	PNN	98.15	37-63	0.32	0.37	97.48	98.54
	CNN	97.46	37-63	0.32	0.37	96.86	97.81
	MLPNN	91.92	37-63	0.32	0.37	91.19	92.34
[137]	F-score+SVM	99.51	80-20	0.35	0.35	100	97.91
[138]	RS_SVM	96.87	80-20	0.35	0.34		
[139]	Hybrid DT	97.85	10-fold CV				
[140]	RS-BPNN	98.6	80-20			98.76	98.57
[141]	RIW-BPNN	99.03	55-45	0.36	0.34	99.13	98.97
	DBN-NN	99.68	55-45	0.33	0.36	100	99.47
[142]	SVM	97.13	10-fold CV				
	C4.5	95.13	10-fold CV				
	Naïve Bayes	95.99	10-fold CV				
	K-NN	95.27	10-fold CV				
[38]	SVM-Convex comb. of kernels	96.79	3-fold CV				
	SVM-Affine comb. of kernels	96.58	3-fold CV				

Ubeyli compares different classifiers (multilayer perceptron neural network (MLPNN), combined neural network (CNN), probabilistic neural network (PNN), recurrent neural network (RNN), support vector machine (SVM)) with respect to their classification accuracies. The author claims that, SVM obtains the highest classification accuracies. For experiments, only one configuration is used, where training set has 80 malignant and 170 benign observations. The rest of the observations are reserved for test set [136].

Akay implements SVM to the breast cancer data and he also applies feature selection. The highest classification accuracies are found to be 98.53%, 99.02% and 99.51% for 50-50%, 70-30%, and 80-20% of training-test partition, respectively. These results are obtained by SVM using five features. Other than classification accuracies, sensitivity, specificity, positive predictive value and negative predictive value of the models are reported. The author reports that, as the size of training set increases, false positive and false negative results decrease [137].

Chen et al. propose a rough set-based SVM (RS_SVM) to diagnose breast cancer. The authors utilize a rough set reduction algorithm for feature selection. As the performance indicators, classification accuracy, sensitivity and specificity are used. The experiments are conducted on three different categories based on training-test set partition: 50-50%, 70-30% and 80-20%. It is concluded that the highest average classifications are obtained with these five features: clump thickness, uniformity of cell shape, marginal adhesion, bare nuclei, and mitoses [138].

Conforti and Guido introduce a kernel-based SVM via semidefinite programming. Their aim is to find the best kernel function for the SVM through an optimization based approach. In their study, they compare the classification accuracies of SVM with convex and affine combinations of kernels. It is observed that, the mean accuracy is almost 97% for both cases [38].

Polat and Gunes conduct a least square SVM (LS-SVM) model and they evaluate it with respect to the classification accuracy, sensitivity and specificity. The experiments are repeated with 50-50%, 70-30% and 80-20% training-test partitions. The authors find that, the classification accuracy of the model is 98.53% [135].

Asri et al. compare the performances of SVM, decision tree, naïve Bayes and K-nearest neighbors (K-NN) on the WBCO dataset [142]. Lavanya and Rani address CART classifier with feature selection and bagging technique. They evaluate the classification performance of the model with respect to accuracy and run time [139].

Abdel-Zaher and Eldeib propose a deep belief network (DBN) unsupervised path followed by back propagation supervised path. Experiments are repeated with three different algorithms: deep belief etwork path (DBN-NN) conjugate gradient back propagation, randomly initialized weight back-propagation neural network (RIW-BPNN) Levenberg-Marquardt and DBN-NN Levenberg-Marquardt. All algorithms achieve classification accuracies higher than 99% [141].

5.2 Machine Learning Applications with the Wisconsin Breast Cancer Diagnostic Dataset

WBCD is another well structured dataset where the breast cancer predictors and the class of tumors (malignant or benign) are included. The number of observations is 569 (212 malignant, 357 benign) and there are 30 predictors under consideration. The predictors are defined as the features that are computed from a digitized image of a fine needle aspirate of a breast mass describing the characteristics of the cell nuclei present in the image. These predictors take values from different scales, thus each factor should be scaled to interval [0,1] to normalize their effects. Table 5.3 shows the predictors of the dataset [6].

Table 5.3: Predictors of the WBCD Dataset

Mean **Radius**	**Radius** Standard Error	Worst **Radius**
Mean **Texture**	**Texture** Standard Error	Worst **Texture**
Mean **Perimeter**	**Perimeter** Standard Error	Worst **Perimeter**
Mean **Area**	**Area** Standard Error	Worst **Area**
Mean **Smoothness**	**Smoothness** Standard Error	Worst **Smoothness**
Mean **Compactness**	**Compactness** Standard Error	Worst **Compactness**
Mean **Concavity**	**Concavity** Standard Error	Worst **Concavity**
Mean **Concave points**	**Concave points** Standard Error	Worst **Concave points**
Mean **Symmetry**	**Symmetry** Standard Error	Worst **Symmetry**
Mean **Fractal dimension**	**Fractal dimension** Standard Error	Worst **Fractal dimension**

Table 5.4 summarizes the studies that work with this dataset. None of these studies consider the case where positive observations are rare in the population. The rarity of the positive observations in the original dataset is 37%, and none of these studies specifies the rarity of positive observations in the training set.

Table 5.4: Various Machine Learning Applications to the WBCD Dataset

Source	Method	Classification Accuracy (%)	Training-Test Partition(%)	Sensitivity (%)	Specificity (%)
[143]	PSO-w/o Feature Selection	96.4	80-20	98.6	93.1
	GA-w/o Feature Selection	96.1		97.8	92.9
	ANN-w/o Feature Selection	96.5		98.2	96
	PSO-with Feature Selection	97.2		98	95.6
	GA-with Feature Selection	96.6		97.5	93.7
	ANN-with Feature Selection	97.3		98.4	95.1
[144]	Linear Regression	96.09	70-30	100	89.8
	MLP	99.04		99.21	98.73
	L1(Manhattan)-NN	93.57		93.46	93.75
	L2(Euclidean)-NN	94.74		97.2	90.63
	Softmax Regression	97.66		100	94.23
	SVM	96.09		97.53	93.62
	GRU-SVM	93.75		100	83.33
[145]	MSM-Tree	97	10-fold CV		
[146]	CART-w/o Feature Selection	92.97			
	CART- SymmetricUncertAttributesetEval	94.72			
[147]	NB	92.97	10-fold CV		
	MLP	96.66			
	J48	93.15			
	SMO	97.72			
	IBK	95.96			
	NB and SMO	97.54			
	MLP and SMO	97.72			
	J48 and SMO	94.09			
	IBK and SMO	97.72			
	SMO, IBK and NB	97.36			
	SMO, IBK and MLP	97.18			
	SMO, IBK and J48	97.36			
	SMO, IBK,NB and MLP	97.54			
	SMO, IBK, NB and J48	97.01			
[139]	CART	92.97	10-fold CV		
	CART with Feature Selection	94.72			
	Hybrid Approach	95.96			
[148]	A-SFM	96.01	5-fold CV		

Aalaei et al. focus on feature selection for diagnosis of breast cancer. The proposed model uses a genetic algorithm based feature selection and it also employs Particle Swarm Optimization algorithm. The authors employ three different classifiers to evaluate the effectiveness of proposed feature selection method: artificial neural network (ANN), particle swarm optimization-based classifier (PSO) and genetic algorithm-based classifier (GA). The experiments are conducted with 80%-20% training-test partition. The results suggest that feature selection improves accuracy. Best accuracy (97.3) is achieved by ANN after feature selection, where the acquired sensitivity and specificity values are 98.4 and 95.1, respectively [143].

Agarap and Fred compare six machine learning algorithms using the WBCD dataset: linear regression, multilayer perceptron (MLP), nearest neighbor search (NN), softmax regression, support vector machine (SVM) and the proposed model, GRU-SVM, which combines a type of recurrent neural network, the Gated Recurrent Unit (GRU), with the support vector machine. 70% and 30% of the data is allocated to training and test sets, respectively [144].

Street et al. address a linear programming-based classification procedure, named MSM-Tree, to find the separating planes for a pattern separation problem. The authors state that MSM-Tree is a variant of the multi-surface method (MSM). 10-fold cross validation accuracy of 97% is achieved with one separating plane and three features: mean texture, worst area and worst smoothness [145] .

Lavanya and Rani analyse the performance of decision tree classifier–CART with and without feature selection on WBCD dataset. According to results, the authors claim that, the best feature selection approach for a particular dataset depends on the number of attributes, attribute type and instances [146].

Salama et al. compare decision tree (J48), multilayer perceptron (MLP), naïve Bayes (NB), sequential minimal optimization (SMO) and instance-based K-nearest neighbor (IBK) using the WBCD dataset. They utilize 10-fold cross validation method and they consider combining multiple classifiers to improve the accuracy. The results show that classification using SMO or fusion of SMO and MLP or fusion of SMO and IBK dominates the other classification methodologies [147].

Fan and Chaovalitwongse propose an optimization framework called support feature machine (SFM) to improve feature selection in medical data classification. The proposed model provides classification and feature selection, simultaneously. SFM works based on voting and averaging schemes. The authors apply their proposed method on different datasets including WBCD. It is observed that, for WBCD dataset, among the proposed models, the highest mean accuracy is achieved with averaging based SFM (96.01%) [148].

Lavanya and Rani address CART classifier with feature selection and bagging technique in classification of tumors. The authors indicate that the best feature selection method for the given dataset is found by evaluating the worth of a feature by measuring the symmetrical uncertainty with respect to the class. The classification accuracies of CART algorithm, CART algorithm with feature selection and a hybrid approach are discussed. Best accuracy is acquired with hybrid approach, which is a combination of the best feature selection method, bagging and decision tree algorithms [139].

Huang and Du [1] and Du and Chen [2] also work on WBCD dataset. Their focus is on classification with uneven training class sizes as it is mentioned in Section 2.4. Remember that, the ratio of positive observations to the whole set of observations is 9% in these studies. Experimental results of of these studies are illustrated in Table 5.5.

Table 5.5: Experimental Results of [1] and [2]

		Accuracy for Benign %	Accuracy for Malignant %	Total Accuracy %
Huang and Du [1]	Standard SVM	94.27	89.1	91.69
	Weighted SVM1	93.63	89.58	91.4
	Weighted SVM2	92.99	90.1	91.4
Du and Chen [2]	Standard V-SVM	98.09	87.5	92.26
	Weighted V-SVM1	97.45	88.54	92.55
	Weighted V-SVM2	96.82	89.06	92.55

5.3 Data

5.3.1 Wisconsin Breast Cancer Original Dataset

After the elimination of the observations with missing values and the removal of correlated factors, a set of 683 observations (239 positive, 444 negative) and 8 factors are found appropriate. The factors are presented in Table 5.6. If there are some predictors whose values are extremely large compared to the others, their values might affect the result more due to larger values even if they are not more important as predictors. However, since all of the factors are in similar ranges, no scaling is necessary. The binary response variable indicates whether a tumor is expected to be malign (1) or not (0).

Table 5.6: Set of Factors: WBCO Dataset

F1	Clump thickness
F2	Uniformity of cell shape
F3	Marginal adhesion
F4	Single epithelial cell
F5	Bare nuclei
F6	Bland chromatin
F7	Normal nuclei
F8	Mitoses

5.3.2 Wisconsin Breast Cancer Diagnostic Dataset

After the elimination of correlations, there remains 21 appropriate factors which are illustrated in Table 5.7.

Table 5.7: Set of Factors: WBCD Dataset

F1	Mean **Radius**	F12	**Concavity** Standard Error
F2	Mean **Texture**	F13	**Concave points** Standard Error
F3	Mean **Smoothness**	F14	**Symmetry** Standard Error
F4	Mean **Compactness**	F15	**Fractal dimension**
F5	Mean **Concavity**	F16	Worst **Smoothness**
F6	Mean **Symmetry**	F17	Worst **Compactness**
F7	Mean **Fractal dimension**	F18	Worst **Concavity**
F8	**Radius** Standard Error	F19	Worst **Concave points**
F9	**Texture** Standard Error	F20	Worst **Symmetry**
F10	**Smoothness** Standard Error	F21	Worst **Fractal dimension**
F11	**Compactness** Standard Error		

We apply feature selection to reduce the number of factors in consistent with the Occam's Razor principle [119]. The purpose of the feature selection is to eliminate redundant features which gives little or no information about the response variable (i.e. type of tumor). Therefore, we first apply t-test on each feature and compare p-value of each feature to measure its effectiveness at group separation. We apply holdout method [149] to the data where the training set size is 400 and test set size is 169. We sort p-values of features in five different instances in order not to favor a specific instance. Then, as another feature selection method, we apply stepwise regression on the same five instances that we have used before. Table 5.8 summarizes the feature selection procedure which employs most significant factors due to both p-value and stepwise regression. First part of Table 5.8 indicates the rank of factors due to p-value for each instance and most significant 10 factors are marked with an asteriks symbol (*). In the second part, factors found significant by stepwise regression for each instance are marked with a star symbol ($\star$). In the third part, the findings of both feature selection methodologies are combined. At the end of the table, final set of selected features are indicated and these factors are shown in Table 5.9, as well.

Table 5.8: Feature Selection (with p-values and stepwise regression)

	F1	**F2**	**F3**	**F4**	**F5**	**F6**	**F7**	**F8**	**F9**	**F10**	**F11**	**F12**	**F13**	**F14**	**F15**	**F16**	**F17**	**F18**	**F19**	**F20**	**F21**
	Feature ranks due to p-values (Most significant 10 features are marked with *)																				
I1	2*	11	13	5*	4*	12	20	7*	19	18	16	14	10*	21	17	8*	6*	3*	1*	9*	15
I2	2*	10*	11	5*	4*	14	20	7*	19	17	15	13	9*	21	18	8*	6*	3*	1*	12	16
I3	2*	8*	11	5*	3*	13	21	7*	19	17	15	16	10*	20	18	9*	6*	4*	1*	12	14
I4	2*	8*	11	5*	3*	13	19	7*	21	18	15	16	10*	20	17	9*	6*	4*	1*	12	14
I5	2*	10*	11	5*	3*	13	21	7*	20	18	15	16	8*	19	17	9*	6*	4*	1*	12	14
	Sequential feature selection- Significant features due to stepwise regression (marked with $\star$)																				
I1	$\star$	$\star$		$\star$	$\star$			$\star$						$\star$		$\star$			$\star$	$\star$	
I2	$\star$	$\star$						$\star$					$\star$						$\star$		
I3	$\star$	$\star$		$\star$	$\star$			$\star$					$\star$						$\star$		
I4	$\star$	$\star$						$\star$							$\star$	$\star$			$\star$		
I5	$\star$	$\star$						$\star$							$\star$	$\star$		$\star$	$\star$		
	Significant features considering both p-values and stepwise regression																				
I1	$\star$*	$\star$		$\star$*	$\star$*			$\star$*					*	$\star$		$\star$*	*	*	$\star$*	$\star$*	
I2	$\star$*	$\star$*		*	*			$\star$*					$\star$*			*	*	*	$\star$*		
I3	$\star$*	$\star$*		$\star$*	$\star$*			$\star$*					$\star$*			*	*	*	$\star$*		
I4	$\star$*	$\star$*		*	*			$\star$*					*		$\star$	$\star$*	*	*	$\star$*		
I5	$\star$*	$\star$*		*	*			$\star$*					*		$\star$	$\star$*	*	$\star$*	$\star$*		
	Selected Features																				
	✓	✓		✓	✓			✓					✓			✓		✓	✓	✓	

Table 5.9: Set of Factors After Feature Selection: WBCD Dataset

Mean **Radius**
Mean **Texture**
Mean **Compactness**
Mean **Concavity**
Radius Standard Error
Concave points Standard Error
Worst **Smoothness**
Worst **Concavity**
Worst **Concave points**
Worst **Symmetry**

Their values are normalized as follows:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}. \tag{5.1}$$

The values are arranged such that, higher values are indication of malignancy. There is no missing information, and the number of observations under consideration is 569 (212 positive, 357 negative).

5.4 Computational Analysis

In this section, we discuss the results of the models PCM+NSGA-II and PCM+RECGA. To analyze the effect of integrating evolutionary algorithms with PCM, first we compare the performances of PCM+NSGA-II and PCM+RECGA with Random+NSGA-II and Random+RECGA, respectively. The computational experiments of these models are conducted with NetBeans IDE 7.3 and CPLEX 12.6 on an Intel(R) Core(TM) i5-2410M 2.3GHz PC with 4 GB RAM, running under the Windows operating system. Next, we compare the performances of PCM+NSGA-II and PCM+RECGA with that of competitor models. They are implemented with the same specifications given in Section 4.3.

We test the model performances by randomly splitting the data into mutually exclusive training ($\mathcal{S}$), validation ($\mathcal{V}$) and test ($\widetilde{\mathcal{S}}$) sets. This procedure is repeated 100 times and their average performances are reported.

Recall that, PCM+NSGA-II and PCM+RECGA first utilize $\mathcal{S}$ to generate the initial set of solutions, and then these solutions are tuned with $\mathcal{V}$. Random+NSGA-II and Random+RECGA just use $\mathcal{V}$ and competitor models utilize $\mathcal{S} \cup \mathcal{V}$ as the training set. Note that, as the training sample grows with the inclusion of the validation set, this provides an advantage to competitor models against PCM+NSGA-II and PCM+RECGA. To test the performance of a model, $\widetilde{\mathcal{S}}$ is used, any time.

To observe the performances of the suggested models in the existence of class imbalance, we create configurations (collection of sets), which simulate different levels of rarity of positive observations.

With this design, we repeat the experiments for the rareness levels ranging between 1% to 35% and 1% to 37% for WBCO dataset and for WBCD dataset, respectively. Note that, number of negative observations is constant, but by eliminating positive observations, rare cases are created.

Table 5.10 and 5.11 show the dataset configurations used in our experiments. The last row of each table indicates the interval of values that the parameter L of PCM can take. Columns "Malign" and "Benign" of Tables 5.10 and 5.11 indicate the number of positive and negative observations in training ($\mathcal{S}$), validation ($\mathcal{V}$) and test samples ($\widetilde{\mathcal{S}}$). "Rareness level" indicates the ratio of positive observations to all observations. The given rareness level is the same both for all observations and for observations that are not used for test purposes. Therefore rareness level is mathematically given as follows: $\frac{|\mathcal{S}^+ \cup \mathcal{V}^+|}{|\mathcal{S} \cup \mathcal{V}|} = \frac{|\mathcal{S}^+ \cup \mathcal{V}^+ \cup \widetilde{\mathcal{S}}^+|}{|\mathcal{S} \cup \mathcal{V} \cup \widetilde{\mathcal{S}}|}$. Note that, while the expression of "low levels of rareness" implies the cases where the ratio of positive observations to all observations is low, the expression "high levels of rareness" refers the opposite.

Table 5.10: Experimental Settings for the WBCO Dataset

	Malign	**Benign**	**Rareness level**		**Malign**	**Benign**	**Rareness level**
$\mathcal{S}$	2	148		$\mathcal{S}$	17	148	
$\mathcal{V}$	2	148		$\mathcal{V}$	17	148	
$\widetilde{\mathcal{S}}$	2	148	**1%**	$\widetilde{\mathcal{S}}$	17	148	**10%**
Total	6	444		Total	51	444	
$\mathcal{S}$	5	148		$\mathcal{S}$	26	148	
$\mathcal{V}$	5	148		$\mathcal{V}$	26	148	
$\widetilde{\mathcal{S}}$	5	148	**3%**	$\widetilde{\mathcal{S}}$	26	148	**15%**
Total	15	444		Total	78	444	
$\mathcal{S}$	8	148		$\mathcal{S}$	49	148	
$\mathcal{V}$	8	148		$\mathcal{V}$	49	148	
$\widetilde{\mathcal{S}}$	8	148	**5%**	$\widetilde{\mathcal{S}}$	49	148	**25%**
Total	24	444		Total	147	444	
$\mathcal{S}$	11	148		$\mathcal{S}$	79	148	
$\mathcal{V}$	11	148		$\mathcal{V}$	79	148	
$\widetilde{\mathcal{S}}$	11	148	**7%**	$\widetilde{\mathcal{S}}$	79	148	**35%**
Total	33	444		Total	237	444	
L for PCM			$\{0,\ldots,148\}$				

Table 5.11: Experimental Settings for the WBCD Dataset

	Malign	**Benign**	**Rareness level**		**Malign**	**Benign**	**Rareness level**
$\mathcal{S}$	1	119		$\mathcal{S}$	13	119	
$\mathcal{V}$	1	119		$\mathcal{V}$	13	119	
$\widetilde{\mathcal{S}}$	1	119	**1%**	$\widetilde{\mathcal{S}}$	13	119	**10%**
Total	3	357		Total	39	357	
$\mathcal{S}$	4	119		$\mathcal{S}$	21	119	
$\mathcal{V}$	4	119		$\mathcal{V}$	21	119	
$\widetilde{\mathcal{S}}$	4	119	**3%**	$\widetilde{\mathcal{S}}$	21	119	**15%**
Total	12	357		Total	63	357	
$\mathcal{S}$	6	119		$\mathcal{S}$	40	119	
$\mathcal{V}$	6	119		$\mathcal{V}$	40	119	
$\widetilde{\mathcal{S}}$	6	119	**5%**	$\widetilde{\mathcal{S}}$	40	119	**25%**
Total	18	357		Total	120	357	
$\mathcal{S}$	9	119		$\mathcal{S}$	70	119	
$\mathcal{V}$	9	119		$\mathcal{V}$	70	119	
$\widetilde{\mathcal{S}}$	9	119	**7%**	$\widetilde{\mathcal{S}}$	70	119	**37%**
Total	27	357		Total	210	357	
L for PCM				$\{0,\ldots,119\}$			

We utilize sensitivity, specificity and Fscore as the main performance indicators. Note that, when occurrence of positive observations in the population are rare, obtaining high true positive rates is more important. High Fscore values are possible only when both of the true positive and true negative classification performances are high. Moreover, we also report accuracy and Fmeasure for the sake of completeness.

Hyper-parameters for the datasets are set due to the previously explained hyper-parameter optimization process. Note that, for both datasets, there are eight different configurations each representing a level of rareness. However, it is computationally expensive to repeat the hyper-parameter optimization process for each rareness level of a dataset. Since it is relatively easy to achieve high prediction results when the classes under consideration are balanced, we focus on the configurations where class

imbalance is significant. Thus, we tune the hyper-parameters using the configurations with 1% and 10% rareness levels in both datasets.

The values stand for hyper-parameters are given in Table 5.12 and 5.13, respectively. Note that, in reporting experimental results, models implemented with the hyper-parameter values corresponding to rareness levels of 1% and 10% will be indicated with $H1$ and $H2$, respectively.

A more detailed explanation of hyper-parameter tuning process for the WBCO and WBCD datasets can be found in the Appendix (Section E and G).

Table 5.12: Optimal Values of Hyper-parameters with Respect to the WBCO Dataset

		Rareness level	
		1%	10%
NSGA-II	*PopulationSize*	1000	500
	GenerationSize	50	5
	NumberOfGenerations	5	5
	p_{rc}, p_{lc}	0.5, 0.5	0.5, 0.5
	p_m	0.01	0.01
RECGA	*PopulationSize*	150	250
	minFinalSetSize	50	200
	p_{rc}, p_{lc}	0.4, 0.6	0.8, 0.2
	p_m	0.01	0.01

Table 5.13: Optimal Values of Hyper-parameters with Respect to the WBCD Dataset

		Rareness level	
		1%	10%
NSGA-II	*PopulationSize*	150	250
	GenerationSize	50	200
	NumberOfGenerations	5	10
	p_{rc}, p_{lc}	0.5, 0.5	0.6, 0.4
	p_m	0.1	0.01
RECGA	*PopulationSize*	1000	250
	minFinalSetSize	50	200
	p_{rc}, p_{lc}	0.0, 1.0	0.5, 0.5
	p_m	0.01	0.01

5.5 Results

5.5.1 Wisconsin Breast Cancer Original Dataset

5.5.1.1 Role of PCM to Generate Initial Solutions to the Evolutionary Algorithms

To evaluate the effect of initial solutions given to the evolutionary algorithms, we compare the performances of the models PCM+NSGA-II, Random+NSGA-II, PCM+RECGA and Random+RECGA. Tables 5.14, 5.15, 5.16 and 5.17 give the training and test performances of these models, respectively.

In the comparison of PCM+NSGA-II and Random+NSGA-II, it is observed that, (regardless of whether the hyper-parameter set is H1 or H2) for all levels of rareness, both models have extremely high specificity performances, which takes values between 0.93 and 1.00. However, in terms of sensitivity values, the performance of the Random+NSGA-II lags behind PCM+NSGA-II. The difference is less in cases where the rareness level is low (1%, 3%, 5%), but as the rareness level increases, the performance of Random+NSGA-II is further impaired. Since Fscore is the harmonic

mean of sensitivity and specificity, the last observation is also true for Fscore performances. The standard deviations of Random+NSGA-II are always higher than that of PCM+NSGA-II, for sensitivity. However, the standard deviation values in specificity are either very close for both models or they are slightly smaller for Random+NSGA-II. These observations are true for both training and test performances. When they are evaluated together, it can be concluded that, it is better to obtain the initial solution set by PCM.

When the same analysis is conducted for PCM+RECGA and Random+RECGA, almost identical observations are obtained. Both in training and test, the specificity values for all levels of rareness are notably high. However, in general, the specificity values are slightly better for Random+RECGA. In training, the sensitivity performances of PCM+RECGA outperforms that of Random+RECGA. The difference in their performances is relatively low when the rareness level is low (1%, 3%, 5%). However, as the rareness level increases, the performance of Random+RECGA decreases. In test, while the sensitivity performance of Random+RECGA is better than PCM+RECGA for rareness levels of 1% and 3%, it lags behind the performance of PCM+RECGA for the remaining configurations (5%, 7%, 10%, 15%, 25% and 35%). The observations given for sensitivity are true for Fscore, as well. In terms of standard deviations, we observed that, either the two models have close values or the standard deviation of Random+RECGA is lower. Thus, although Random+RECGA seems to achieve good performances when the rareness level is very low, its performance deteriorates significantly for slightly higher values of rareness (starting from 5%).

Therefore, we can conclude that, in general, generating initial solutions of the RECGA algorithm with PCM yields better results.

Table 5.14: Average Training Performances: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA (WBCO Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level = 1%							
Sensitivity	0.93	0.70	0.88	0.81	0.92	0.93	0.87	0.87
Specificity	0.93	0.98	0.95	0.99	0.96	0.97	0.99	0.99
Accuracy	0.93	0.98	0.95	0.98	0.96	0.97	0.98	0.98
Fscore	0.92	0.79	0.89	0.87	0.92	0.94	0.90	0.90
Fmeasure	0.31	0.66	0.38	0.59	0.56	0.59	0.62	0.62
	Rareness level = 3%							
Sensitivity	0.86	0.82	0.80	0.81	0.86	0.89	0.81	0.81
Specificity	0.96	1.00	0.98	0.99	0.97	0.97	0.99	0.99
Accuracy	0.96	0.99	0.97	0.99	0.97	0.97	0.98	0.98
Fscore	0.90	0.89	0.87	0.88	0.90	0.92	0.88	0.88
Fmeasure	0.62	0.84	0.67	0.81	0.68	0.70	0.77	0.77
	Rareness level = 5%							
Sensitivity	0.77	0.84	0.66	0.76	0.82	0.86	0.70	0.70
Specificity	0.98	0.99	0.98	0.99	0.96	0.97	0.99	0.99
Accuracy	0.97	0.98	0.97	0.98	0.95	0.96	0.98	0.98
Fscore	0.86	0.91	0.78	0.85	0.88	0.90	0.81	0.81
Fmeasure	0.74	0.84	0.68	0.80	0.69	0.72	0.77	0.77
	Rareness level = 7%							
Sensitivity	0.80	0.88	0.67	0.79	0.85	0.88	0.72	0.72
Specificity	0.99	0.99	0.99	0.99	0.96	0.96	1.00	1.00
Accuracy	0.97	0.98	0.97	0.98	0.96	0.96	0.98	0.98
Fscore	0.88	0.93	0.79	0.87	0.90	0.91	0.83	0.83
Fmeasure	0.80	0.89	0.72	0.84	0.75	0.76	0.81	0.81
	Rareness level = 10%							
Sensitivity	0.84	0.90	0.64	0.79	0.84	0.87	0.68	0.68
Specificity	0.99	0.99	0.99	1.00	0.97	0.97	1.00	1.00
Accuracy	0.97	0.98	0.96	0.97	0.96	0.96	0.97	0.97
Fscore	0.90	0.94	0.77	0.88	0.89	0.91	0.80	0.80
Fmeasure	0.85	0.91	0.74	0.86	0.80	0.81	0.80	0.80
	Rareness level = 15%							
Sensitivity	0.89	0.92	0.62	0.79	0.86	0.87	0.68	0.68
Specificity	0.98	0.99	1.00	1.00	0.96	0.97	1.00	1.00
Accuracy	0.97	0.98	0.94	0.96	0.95	0.95	0.95	0.95
Fscore	0.93	0.95	0.76	0.87	0.90	0.91	0.81	0.81
Fmeasure	0.90	0.93	0.75	0.87	0.83	0.84	0.80	0.80
	Rareness level = 25%							
Sensitivity	0.94	0.93	0.61	0.77	0.85	0.86	0.62	0.62
Specificity	0.98	0.99	1.00	1.00	0.97	0.97	1.00	1.00
Accuracy	0.97	0.97	0.90	0.94	0.94	0.94	0.90	0.90
Fscore	0.96	0.96	0.75	0.86	0.89	0.90	0.76	0.76
Fmeasure	0.94	0.94	0.75	0.86	0.86	0.87	0.76	0.76
	Rareness level = 35%							
Sensitivity	0.96	0.94	0.66	0.78	0.90	0.91	0.64	0.64
Specificity	0.98	0.98	1.00	0.99	0.96	0.96	1.00	1.00
Accuracy	0.97	0.97	0.88	0.92	0.94	0.94	0.87	0.87
Fscore	0.97	0.96	0.79	0.87	0.92	0.93	0.78	0.78
Fmeasure	0.96	0.95	0.79	0.87	0.91	0.91	0.78	0.78

Table 5.15: Standard Deviations of Training Performance Indicators: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA (WBCO Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level = 1%							
Sensitivity	0.17	0.25	0.24	0.23	0.17	0.19	0.22	0.22
Specificity	0.04	0.10	0.01	0.04	0.05	0.07	0.01	0.01
Accuracy	0.03	0.10	0.01	0.04	0.05	0.07	0.01	0.01
Fscore	0.11	0.18	0.16	0.16	0.11	0.12	0.15	0.15
Fmeasure	0.14	0.19	0.21	0.17	0.28	0.28	0.20	0.20
	Rareness level = 3%							
Sensitivity	0.14	0.11	0.14	0.17	0.15	0.16	0.15	0.15
Specificity	0.03	0.00	0.01	0.02	0.03	0.03	0.01	0.01
Accuracy	0.02	0.01	0.01	0.02	0.03	0.03	0.01	0.01
Fscore	0.08	0.07	0.09	0.11	0.08	0.09	0.09	0.09
Fmeasure	0.12	0.09	0.11	0.13	0.16	0.17	0.12	0.12
	Rareness level = 5%							
Sensitivity	0.14	0.11	0.15	0.17	0.14	0.15	0.16	0.16
Specificity	0.01	0.02	0.01	0.01	0.04	0.05	0.01	0.01
Accuracy	0.01	0.02	0.01	0.01	0.04	0.04	0.01	0.01
Fscore	0.09	0.07	0.11	0.13	0.08	0.09	0.12	0.12
Fmeasure	0.10	0.11	0.11	0.11	0.12	0.13	0.12	0.12
	Rareness level = 7%							
Sensitivity	0.10	0.08	0.13	0.15	0.14	0.15	0.14	0.14
Specificity	0.01	0.01	0.01	0.01	0.04	0.04	0.00	0.00
Accuracy	0.01	0.01	0.01	0.01	0.03	0.03	0.01	0.01
Fscore	0.06	0.05	0.09	0.11	0.09	0.09	0.10	0.10
Fmeasure	0.07	0.06	0.09	0.09	0.13	0.12	0.10	0.10
	Rareness level = 10%							
Sensitivity	0.08	0.07	0.11	0.12	0.14	0.15	0.12	0.12
Specificity	0.01	0.01	0.01	0.01	0.03	0.03	0.00	0.00
Accuracy	0.01	0.01	0.01	0.01	0.03	0.03	0.01	0.01
Fscore	0.05	0.04	0.07	0.09	0.09	0.10	0.09	0.09
Fmeasure	0.06	0.06	0.07	0.08	0.10	0.10	0.09	0.09
	Rareness level = 15%							
Sensitivity	0.06	0.07	0.11	0.11	0.16	0.17	0.09	0.09
Specificity	0.01	0.01	0.00	0.01	0.03	0.06	0.00	0.00
Accuracy	0.01	0.01	0.02	0.01	0.02	0.05	0.01	0.01
Fscore	0.03	0.05	0.07	0.09	0.10	0.11	0.07	0.07
Fmeasure	0.04	0.05	0.07	0.08	0.09	0.11	0.07	0.07
	Rareness level = 25%							
Sensitivity	0.03	0.07	0.12	0.09	0.18	0.18	0.07	0.07
Specificity	0.01	0.01	0.00	0.00	0.03	0.04	0.00	0.00
Accuracy	0.01	0.02	0.03	0.02	0.04	0.04	0.02	0.02
Fscore	0.02	0.04	0.09	0.07	0.12	0.12	0.06	0.06
Fmeasure	0.02	0.04	0.09	0.07	0.12	0.12	0.06	0.06
	Rareness level = 35%							
Sensitivity	0.02	0.07	0.05	0.05	0.13	0.12	0.11	0.06
Specificity	0.01	0.03	0.00	0.00	0.03	0.03	0.01	0.00
Accuracy	0.01	0.03	0.02	0.02	0.04	0.04	0.04	0.02
Fscore	0.01	0.04	0.04	0.04	0.08	0.08	0.07	0.05
Fmeasure	0.01	0.04	0.04	0.04	0.08	0.08	0.07	0.05

Table 5.16: Average Test Performances: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA (WBCO Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level = 1%							
Sensitivity	0.93	0.65	0.81	0.73	0.74	0.73	0.84	0.84
Specificity	0.93	0.98	0.95	0.98	0.95	0.96	0.98	0.98
Accuracy	0.93	0.97	0.94	0.97	0.95	0.96	0.98	0.98
Fscore	0.91	0.73	0.83	0.80	0.76	0.76	0.88	0.88
Fmeasure	0.31	0.54	0.34	0.46	0.40	0.42	0.51	0.51
	Rareness level =3%							
Sensitivity	0.85	0.67	0.81	0.74	0.75	0.76	0.82	0.82
Specificity	0.96	0.99	0.98	0.99	0.97	0.97	0.99	0.99
Accuracy	0.96	0.98	0.97	0.98	0.96	0.96	0.98	0.98
Fscore	0.89	0.77	0.87	0.83	0.82	0.83	0.89	0.89
Fmeasure	0.61	0.64	0.67	0.71	0.59	0.60	0.75	0.75
	Rareness level =5%							
Sensitivity	0.82	0.76	0.79	0.75	0.79	0.80	0.77	0.77
Specificity	0.98	0.98	0.98	0.99	0.96	0.96	0.99	0.99
Accuracy	0.97	0.97	0.97	0.98	0.95	0.95	0.98	0.98
Fscore	0.89	0.84	0.87	0.85	0.85	0.86	0.86	0.86
Fmeasure	0.76	0.72	0.77	0.77	0.65	0.66	0.81	0.81
	Rareness level =7%							
Sensitivity	0.78	0.76	0.68	0.70	0.80	0.82	0.69	0.69
Specificity	0.98	0.98	0.99	0.99	0.96	0.96	1.00	1.00
Accuracy	0.97	0.97	0.97	0.97	0.95	0.95	0.97	0.97
Fscore	0.86	0.84	0.79	0.81	0.85	0.87	0.80	0.80
Fmeasure	0.77	0.75	0.73	0.76	0.69	0.70	0.78	0.78
	Rareness level =10%							
Sensitivity	0.78	0.80	0.56	0.64	0.79	0.81	0.62	0.62
Specificity	0.98	0.98	0.99	0.99	0.97	0.96	1.00	1.00
Accuracy	0.96	0.96	0.95	0.96	0.95	0.95	0.96	0.96
Fscore	0.86	0.87	0.70	0.77	0.85	0.87	0.76	0.76
Fmeasure	0.80	0.80	0.67	0.74	0.76	0.76	0.75	0.75
	Rareness level =15%							
Sensitivity	0.84	0.84	0.57	0.68	0.82	0.83	0.64	0.64
Specificity	0.98	0.98	1.00	0.99	0.96	0.96	1.00	1.00
Accuracy	0.96	0.96	0.93	0.95	0.94	0.94	0.95	0.95
Fscore	0.90	0.90	0.72	0.80	0.87	0.88	0.78	0.78
Fmeasure	0.86	0.85	0.71	0.79	0.80	0.81	0.77	0.77
	Rareness level =25%							
Sensitivity	0.93	0.90	0.63	0.74	0.86	0.86	0.68	0.68
Specificity	0.97	0.97	1.00	0.99	0.96	0.96	1.00	1.00
Accuracy	0.96	0.96	0.90	0.93	0.93	0.94	0.92	0.92
Fscore	0.95	0.93	0.77	0.84	0.89	0.90	0.81	0.81
Fmeasure	0.92	0.91	0.76	0.83	0.86	0.86	0.81	0.81
	Rareness level =35%							
Sensitivity	0.94	0.91	0.61	0.73	0.91	0.91	0.62	0.62
Specificity	0.97	0.97	1.00	0.99	0.95	0.95	1.00	1.00
Accuracy	0.96	0.95	0.86	0.90	0.93	0.93	0.87	0.87
Fscore	0.96	0.94	0.75	0.83	0.92	0.92	0.76	0.76
Fmeasure	0.94	0.92	0.75	0.83	0.90	0.90	0.76	0.76

Table 5.17: Standard Deviations of Test Performance Indicators: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA (WBCO Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level =1%							
Sensitivity	0.19	0.33	0.30	0.28	0.34	0.34	0.25	0.25
Specificity	0.05	0.10	0.01	0.05	0.05	0.08	0.01	0.01
Accuracy	0.05	0.10	0.01	0.05	0.05	0.08	0.01	0.01
Fscore	0.14	0.30	0.24	0.22	0.31	0.31	0.19	0.19
Fmeasure	0.16	0.26	0.23	0.19	0.26	0.25	0.18	0.18
	Rareness level =3%							
Sensitivity	0.19	0.23	0.20	0.18	0.23	0.23	0.18	0.18
Specificity	0.03	0.01	0.01	0.02	0.03	0.04	0.01	0.01
Accuracy	0.03	0.01	0.01	0.02	0.03	0.03	0.01	0.01
Fscore	0.12	0.19	0.15	0.11	0.18	0.18	0.12	0.12
Fmeasure	0.14	0.18	0.15	0.14	0.17	0.18	0.13	0.13
	Rareness level =5%							
Sensitivity	0.14	0.18	0.15	0.15	0.18	0.19	0.14	0.14
Specificity	0.02	0.02	0.01	0.02	0.05	0.06	0.01	0.01
Accuracy	0.01	0.02	0.01	0.02	0.04	0.05	0.01	0.01
Fscore	0.09	0.12	0.11	0.10	0.12	0.13	0.09	0.09
Fmeasure	0.10	0.13	0.11	0.12	0.13	0.15	0.10	0.10
	Rareness level =7%							
Sensitivity	0.15	0.16	0.16	0.18	0.18	0.19	0.15	0.15
Specificity	0.01	0.01	0.01	0.01	0.05	0.04	0.01	0.01
Accuracy	0.01	0.01	0.01	0.01	0.04	0.04	0.01	0.01
Fscore	0.10	0.11	0.11	0.13	0.12	0.12	0.11	0.11
Fmeasure	0.09	0.11	0.10	0.12	0.13	0.13	0.11	0.11
	Rareness level =10%							
Sensitivity	0.12	0.14	0.14	0.17	0.18	0.19	0.12	0.12
Specificity	0.01	0.01	0.01	0.01	0.03	0.03	0.00	0.00
Accuracy	0.01	0.02	0.01	0.01	0.03	0.03	0.01	0.01
Fscore	0.08	0.09	0.11	0.14	0.12	0.13	0.10	0.10
Fmeasure	0.08	0.09	0.11	0.13	0.11	0.12	0.10	0.10
	Rareness level =15%							
Sensitivity	0.09	0.10	0.11	0.12	0.16	0.16	0.10	0.10
Specificity	0.01	0.01	0.01	0.01	0.03	0.06	0.00	0.00
Accuracy	0.01	0.02	0.02	0.02	0.03	0.05	0.02	0.02
Fscore	0.05	0.06	0.08	0.10	0.10	0.11	0.08	0.08
Fmeasure	0.06	0.06	0.08	0.10	0.09	0.11	0.08	0.08
	Rareness level =25%							
Sensitivity	0.05	0.07	0.12	0.11	0.17	0.17	0.06	0.06
Specificity	0.02	0.02	0.01	0.00	0.03	0.04	0.00	0.00
Accuracy	0.01	0.02	0.03	0.03	0.04	0.04	0.02	0.02
Fscore	0.02	0.04	0.09	0.09	0.11	0.11	0.04	0.04
Fmeasure	0.03	0.03	0.09	0.09	0.10	0.10	0.04	0.04
	Rareness level =35%							
Sensitivity	0.04	0.08	0.12	0.08	0.14	0.14	0.05	0.05
Specificity	0.02	0.03	0.01	0.00	0.04	0.04	0.00	0.00
Accuracy	0.01	0.03	0.04	0.03	0.04	0.04	0.02	0.02
Fscore	0.02	0.05	0.09	0.07	0.09	0.09	0.04	0.04
Fmeasure	0.02	0.05	0.09	0.07	0.09	0.09	0.04	0.04

Table 5.18 gives the solution times of PCM+NSGA-II, Random+NSGA-II, PCM+RECGA and Random+RECGA.

Table 5.18: Solution Times (in sec.) (WBCO Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level =1%							
AVG	12.37	8.80	8.64	4.28	4.59	7.87	0.99	1.61
STD.DEV	0.51	0.61	0.17	0.22	1.31	0.42	0.11	0.07
	Rareness level =3%							
AVG	12.44	8.89	8.69	4.36	4.57	6.52	1.03	1.64
STD.DEV	0.25	0.27	0.09	0.17	0.70	0.20	0.06	0.08
	Rareness level =5%							
AVG	12.73	9.08	8.87	4.46	4.76	6.39	1.10	1.77
STD.DEV	0.27	0.32	0.09	0.13	0.36	0.23	0.09	0.07
	Rareness level =7%							
AVG	13.05	13.14	9.01	4.58	4.97	10.53	1.11	1.80
STD.DEV	0.55	14.13	0.07	0.16	14.42	0.47	0.06	0.03
	Rareness level =10%							
AVG	15.17	58.75	9.28	4.79	6.55	56.39	1.18	1.92
STD.DEV	0.73	19.17	0.09	0.18	19.28	0.67	0.07	0.04
	Rareness level =15%							
AVG	16.36	67.18	9.82	5.11	7.26	64.78	1.25	2.06
STD.DEV	0.33	2.20	0.11	0.15	1.94	0.25	0.07	0.04
	Rareness level =25%							
AVG	18.57	69.49	11.07	5.85	8.47	66.94	1.47	2.44
STD.DEV	0.48	2.31	0.09	0.14	1.64	0.31	0.06	0.04
	Rareness level =35%							
AVG	21.32	73.63	12.54	6.78	10.06	69.25	1.75	2.90
STD.DEV	0.63	6.53	0.11	0.15	1.90	0.50	0.07	0.04

5.5.1.2 Comparison of PCM+NSGA-II, PCM+RECGA and Competitor Models

In this section, we compare the performances of PCM+NSGA-II and PCM+RECGA with those of competitor models. Tables 5.19, 5.21, 5.20 and 5.22 summarize average performances and standard deviations of performance indicators for training and test, respectively.

Figure 5.1 illustrates the training and test performances of the models based on the results given in Tables 5.19 and 5.21. Figure 5.2 shows the gaps between the models' training and test performances. Note that, the gap refers to how much the training performance is greater than the test performance and if the training performance lags behind the test, the gap is expressed as zero.

Table 5.19: Average Training Performances (WBCO Dataset)

	PCM+NSGA-II^{H1}	PCM+NSGA-II^{H2}	PCM+RECGA^{H1}	PCM+RECGA^{H2}	LR	pen-LR	SVM	ANN	DT	RF
	Rareness level =1%									
Sensitivity	0.93	0.70	0.92	0.93	0.97	0.94	0.69	0.96	0.84	1.00
Specificity	0.93	0.98	0.96	0.97	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	0.93	0.98	0.96	0.97	1.00	1.00	1.00	1.00	0.99	1.00
Fscore	0.92	0.79	0.92	0.94	0.98	0.96	0.71	0.97	0.87	1.00
Fmeasure	0.31	0.66	0.56	0.59	0.98	0.95	0.71	0.97	0.79	1.00
	Rareness level =3%									
Sensitivity	0.86	0.82	0.86	0.89	0.91	0.88	0.79	0.97	0.87	1.00
Specificity	0.96	1.00	0.97	0.97	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	0.96	0.99	0.97	0.97	0.99	0.99	0.99	1.00	0.99	1.00
Fscore	0.90	0.89	0.90	0.92	0.95	0.93	0.86	0.98	0.92	1.00
Fmeasure	0.62	0.84	0.68	0.70	0.92	0.90	0.83	0.97	0.87	1.00
	Rareness level =5%									
Sensitivity	0.77	0.84	0.82	0.86	0.89	0.87	0.83	0.97	0.86	1.00
Specificity	0.98	0.99	0.96	0.97	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	0.97	0.98	0.95	0.96	0.99	0.99	0.99	1.00	0.99	1.00
Fscore	0.86	0.91	0.88	0.90	0.94	0.93	0.90	0.98	0.92	1.00
Fmeasure	0.74	0.84	0.69	0.72	0.90	0.89	0.86	0.96	0.89	1.00
	Rareness level =7%									
Sensitivity	0.80	0.88	0.85	0.88	0.90	0.89	0.88	0.98	0.90	1.00
Specificity	0.99	0.99	0.96	0.96	0.99	0.99	0.99	1.00	0.99	1.00
Accuracy	0.97	0.98	0.96	0.96	0.99	0.99	0.98	1.00	0.99	1.00
Fscore	0.88	0.93	0.90	0.91	0.94	0.94	0.93	0.99	0.94	1.00
Fmeasure	0.80	0.89	0.75	0.76	0.91	0.90	0.89	0.97	0.90	1.00
	Rareness level =10%									
Sensitivity	0.84	0.90	0.84	0.87	0.92	0.91	0.91	0.99	0.93	1.00
Specificity	0.99	0.99	0.97	0.97	0.99	0.99	0.99	0.99	0.99	1.00
Accuracy	0.97	0.98	0.96	0.96	0.98	0.98	0.98	0.99	0.98	1.00
Fscore	0.90	0.94	0.89	0.91	0.95	0.95	0.95	0.99	0.96	1.00
Fmeasure	0.85	0.91	0.80	0.81	0.92	0.92	0.91	0.97	0.93	1.00
	Rareness level =15%									
Sensitivity	0.89	0.92	0.86	0.87	0.93	0.92	0.94	0.99	0.95	1.00
Specificity	0.98	0.99	0.96	0.97	0.99	0.99	0.99	0.99	0.99	1.00
Accuracy	0.97	0.98	0.95	0.95	0.98	0.98	0.98	0.99	0.98	1.00
Fscore	0.93	0.95	0.90	0.91	0.96	0.95	0.96	0.99	0.97	1.00
Fmeasure	0.90	0.93	0.83	0.84	0.93	0.93	0.93	0.97	0.94	1.00
	Rareness level =25%									
Sensitivity	0.94	0.93	0.85	0.86	0.95	0.94	0.96	1.00	0.97	1.00
Specificity	0.98	0.99	0.97	0.97	0.98	0.98	0.98	0.98	0.98	1.00
Accuracy	0.97	0.97	0.94	0.94	0.97	0.97	0.98	0.99	0.98	1.00
Fscore	0.96	0.96	0.89	0.90	0.96	0.96	0.97	0.99	0.98	1.00
Fmeasure	0.94	0.94	0.86	0.87	0.95	0.94	0.95	0.98	0.96	1.00
	Rareness level =35%									
Sensitivity	0.96	0.94	0.90	0.91	0.96	0.96	0.97	1.00	0.98	1.00
Specificity	0.98	0.98	0.96	0.96	0.98	0.98	0.98	0.98	0.98	1.00
Accuracy	0.97	0.97	0.94	0.94	0.97	0.97	0.97	0.99	0.98	1.00
Fscore	0.97	0.96	0.92	0.93	0.97	0.97	0.97	0.99	0.98	1.00
Fmeasure	0.96	0.95	0.91	0.91	0.96	0.96	0.96	0.98	0.97	1.00

Table 5.20: Standard Deviations of Training Performance Indicators (WBCO Dataset)

	PCM+NSGA-II ^{H1}	PCM+NSGA-II ^{H2}	PCM+RECGA ^{H1}	PCM+RECGA ^{H2}	LR	pen-LR	SVM	ANN	DT	RF
	Rareness level =1%									
Sensitivity	0.17	0.25	0.17	0.19	0.10	0.14	0.41	0.12	0.27	0.00
Specificity	0.04	0.10	0.05	0.07	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.03	0.10	0.05	0.07	0.00	0.00	0.01	0.00	0.00	0.00
Fscore	0.11	0.18	0.11	0.12	0.07	0.09	0.41	0.08	0.24	0.00
Fmeasure	0.14	0.19	0.28	0.28	0.08	0.10	0.41	0.09	0.22	0.00
	Rareness level =3%									
Sensitivity	0.14	0.11	0.15	0.16	0.10	0.11	0.23	0.05	0.12	0.00
Specificity	0.03	0.00	0.03	0.03	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.02	0.01	0.03	0.03	0.01	0.01	0.01	0.00	0.00	0.00
Fscore	0.08	0.07	0.08	0.09	0.06	0.06	0.21	0.03	0.07	0.00
Fmeasure	0.12	0.09	0.16	0.17	0.09	0.09	0.21	0.04	0.07	0.00
	Rareness level =5%									
Sensitivity	0.14	0.11	0.14	0.15	0.09	0.08	0.15	0.05	0.10	0.00
Specificity	0.01	0.02	0.04	0.05	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.01	0.02	0.04	0.04	0.01	0.01	0.01	0.00	0.00	0.00
Fscore	0.09	0.07	0.08	0.09	0.05	0.05	0.13	0.03	0.06	0.00
Fmeasure	0.10	0.11	0.12	0.13	0.08	0.07	0.13	0.04	0.05	0.00
	Rareness level =7%									
Sensitivity	0.10	0.08	0.14	0.15	0.06	0.06	0.08	0.04	0.07	0.00
Specificity	0.01	0.01	0.04	0.04	0.00	0.00	0.00	0.00	0.01	0.00
Accuracy	0.01	0.01	0.03	0.03	0.01	0.01	0.01	0.01	0.00	0.00
Fscore	0.06	0.05	0.09	0.09	0.04	0.04	0.05	0.02	0.04	0.00
Fmeasure	0.07	0.06	0.13	0.12	0.05	0.05	0.06	0.04	0.04	0.00
	Rareness level =10%									
Sensitivity	0.08	0.07	0.14	0.15	0.04	0.04	0.06	0.03	0.04	0.00
Specificity	0.01	0.01	0.03	0.03	0.00	0.00	0.01	0.01	0.01	0.00
Accuracy	0.01	0.01	0.03	0.03	0.01	0.01	0.01	0.01	0.01	0.00
Fscore	0.05	0.04	0.09	0.10	0.02	0.02	0.03	0.01	0.02	0.00
Fmeasure	0.06	0.06	0.10	0.10	0.04	0.04	0.05	0.03	0.03	0.00
	Rareness level =15%									
Sensitivity	0.06	0.07	0.16	0.17	0.03	0.03	0.04	0.02	0.03	0.00
Specificity	0.01	0.01	0.03	0.06	0.00	0.00	0.01	0.01	0.01	0.00
Accuracy	0.01	0.01	0.02	0.05	0.01	0.01	0.01	0.01	0.01	0.00
Fscore	0.03	0.05	0.10	0.11	0.02	0.02	0.02	0.01	0.02	0.00
Fmeasure	0.04	0.05	0.09	0.11	0.02	0.02	0.03	0.02	0.02	0.00
	Rareness level =25%									
Sensitivity	0.03	0.07	0.18	0.18	0.01	0.02	0.02	0.01	0.02	0.00
Specificity	0.01	0.01	0.03	0.04	0.00	0.00	0.01	0.01	0.01	0.00
Accuracy	0.01	0.02	0.04	0.04	0.01	0.01	0.01	0.01	0.01	0.00
Fscore	0.02	0.04	0.12	0.12	0.01	0.01	0.01	0.01	0.01	0.00
Fmeasure	0.02	0.04	0.12	0.12	0.01	0.01	0.02	0.01	0.01	0.00
	Rareness level =35%									
Sensitivity	0.02	0.07	0.13	0.12	0.01	0.01	0.01	0.00	0.01	0.00
Specificity	0.01	0.03	0.03	0.03	0.00	0.00	0.01	0.01	0.01	0.00
Accuracy	0.01	0.03	0.04	0.04	0.01	0.01	0.00	0.00	0.01	0.00
Fscore	0.01	0.04	0.08	0.08	0.01	0.01	0.01	0.00	0.01	0.00
Fmeasure	0.01	0.04	0.08	0.08	0.01	0.01	0.01	0.01	0.01	0.00

Table 5.21: Average Test Performances (WBCO Dataset)

	PCM+NSGA-II^{H1}	PCM+NSGA-II^{H2}	PCM+RECGA^{H1}	PCM+RECGA^{H2}	LR	pen-LR	SVM	ANN	DT	RF
Rareness level =1%										
Sensitivity	0.93	0.65	0.74	0.73	0.50	0.52	0.29	0.58	0.47	0.46
Specificity	0.93	0.98	0.95	0.96	0.99	0.99	1.00	0.99	0.99	0.99
Accuracy	0.93	0.97	0.95	0.96	0.98	0.99	0.99	0.99	0.99	0.99
Fscore	0.91	0.73	0.76	0.76	0.57	0.61	0.32	0.64	0.54	0.53
Fmeasure	0.31	0.54	0.40	0.42	0.42	0.46	0.28	0.52	0.42	0.43
Rareness level =3%										
Sensitivity	0.85	0.67	0.75	0.76	0.63	0.63	0.56	0.64	0.63	0.63
Specificity	0.96	0.99	0.97	0.97	0.99	0.99	0.99	0.99	0.98	0.99
Accuracy	0.96	0.98	0.96	0.96	0.98	0.98	0.98	0.98	0.97	0.98
Fscore	0.89	0.77	0.82	0.83	0.74	0.75	0.66	0.74	0.74	0.74
Fmeasure	0.61	0.64	0.59	0.60	0.63	0.64	0.60	0.62	0.59	0.62
Rareness level =5%										
Sensitivity	0.82	0.76	0.79	0.80	0.73	0.73	0.71	0.71	0.66	0.70
Specificity	0.98	0.98	0.96	0.96	0.99	0.99	0.99	0.98	0.98	0.98
Accuracy	0.97	0.97	0.95	0.95	0.97	0.97	0.98	0.97	0.97	0.97
Fscore	0.89	0.84	0.85	0.86	0.83	0.83	0.80	0.81	0.77	0.80
Fmeasure	0.76	0.72	0.65	0.66	0.73	0.73	0.73	0.69	0.65	0.69
Rareness level =7%										
Sensitivity	0.78	0.76	0.80	0.82	0.75	0.73	0.76	0.73	0.68	0.75
Specificity	0.98	0.98	0.96	0.96	0.98	0.99	0.99	0.98	0.98	0.98
Accuracy	0.97	0.97	0.95	0.95	0.97	0.97	0.97	0.96	0.96	0.96
Fscore	0.86	0.84	0.85	0.87	0.84	0.83	0.85	0.82	0.79	0.84
Fmeasure	0.77	0.75	0.69	0.70	0.76	0.75	0.78	0.73	0.69	0.74
Rareness level =10%										
Sensitivity	0.78	0.80	0.79	0.81	0.81	0.79	0.83	0.82	0.78	0.80
Specificity	0.98	0.98	0.97	0.96	0.98	0.98	0.98	0.97	0.97	0.98
Accuracy	0.96	0.96	0.95	0.95	0.97	0.96	0.97	0.96	0.95	0.96
Fscore	0.86	0.87	0.85	0.87	0.89	0.87	0.90	0.88	0.86	0.87
Fmeasure	0.80	0.80	0.76	0.76	0.83	0.82	0.84	0.80	0.77	0.80
Rareness level =15%										
Sensitivity	0.84	0.84	0.82	0.83	0.88	0.87	0.91	0.89	0.83	0.89
Specificity	0.98	0.98	0.96	0.96	0.98	0.98	0.98	0.97	0.97	0.98
Accuracy	0.96	0.96	0.94	0.94	0.97	0.96	0.97	0.96	0.95	0.96
Fscore	0.90	0.90	0.87	0.88	0.93	0.92	0.94	0.93	0.89	0.93
Fmeasure	0.86	0.85	0.80	0.81	0.89	0.88	0.89	0.87	0.82	0.88
Rareness level =25%										
Sensitivity	0.93	0.90	0.86	0.86	0.94	0.94	0.96	0.95	0.92	0.95
Specificity	0.97	0.97	0.96	0.96	0.98	0.98	0.97	0.96	0.96	0.97
Accuracy	0.96	0.96	0.93	0.94	0.97	0.97	0.97	0.96	0.95	0.97
Fscore	0.95	0.93	0.89	0.90	0.96	0.96	0.96	0.96	0.94	0.96
Fmeasure	0.92	0.91	0.86	0.86	0.94	0.93	0.94	0.92	0.90	0.93
Rareness level =35%										
Sensitivity	0.94	0.91	0.91	0.91	0.95	0.95	0.96	0.97	0.93	0.97
Specificity	0.97	0.97	0.95	0.95	0.97	0.97	0.97	0.96	0.95	0.96
Accuracy	0.96	0.95	0.93	0.93	0.97	0.97	0.97	0.96	0.95	0.97
Fscore	0.96	0.94	0.92	0.92	0.96	0.96	0.97	0.96	0.94	0.97
Fmeasure	0.94	0.92	0.90	0.90	0.95	0.95	0.95	0.95	0.92	0.95

Table 5.22: Standard Deviations of Test Performance Indicators (WBCO Dataset)

	PCM+NSGA-II^{H1}	PCM+NSGA-II^{H2}	PCM+RECGA ^{H1}	PCM+RECGA ^{H2}	LR	pen-LR	SVM	ANN	DT	RF
Rareness level =1%										
Sensitivity	0.19	0.33	0.34	0.34	0.37	0.34	0.39	0.37	0.37	0.36
Specificity	0.05	0.10	0.05	0.08	0.01	0.01	0.00	0.01	0.01	0.01
Accuracy	0.05	0.10	0.05	0.08	0.01	0.01	0.00	0.01	0.01	0.01
Fscore	0.14	0.30	0.31	0.31	0.38	0.34	0.42	0.36	0.38	0.38
Fmeasure	0.16	0.26	0.26	0.25	0.31	0.28	0.36	0.31	0.32	0.33
Rareness level =3%										
Sensitivity	0.19	0.23	0.23	0.23	0.23	0.22	0.30	0.25	0.22	0.23
Specificity	0.03	0.01	0.03	0.04	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.03	0.01	0.03	0.03	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.12	0.19	0.18	0.18	0.20	0.18	0.30	0.22	0.19	0.19
Fmeasure	0.14	0.18	0.17	0.18	0.19	0.18	0.27	0.19	0.17	0.19
Rareness level =5%										
Sensitivity	0.14	0.18	0.18	0.19	0.17	0.16	0.21	0.20	0.20	0.19
Specificity	0.02	0.02	0.05	0.06	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.01	0.02	0.04	0.05	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.09	0.12	0.12	0.13	0.13	0.12	0.19	0.15	0.16	0.14
Fmeasure	0.10	0.13	0.13	0.15	0.13	0.12	0.18	0.14	0.14	0.14
Rareness level =7%										
Sensitivity	0.15	0.16	0.18	0.19	0.15	0.16	0.14	0.18	0.15	0.14
Specificity	0.01	0.01	0.05	0.04	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.01	0.01	0.04	0.04	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.10	0.11	0.12	0.12	0.10	0.11	0.12	0.14	0.11	0.09
Fmeasure	0.09	0.11	0.13	0.13	0.11	0.11	0.11	0.13	0.10	0.10
Rareness level =10%										
Sensitivity	0.12	0.14	0.18	0.19	0.09	0.10	0.09	0.13	0.11	0.11
Specificity	0.01	0.01	0.03	0.03	0.01	0.01	0.01	0.01	0.02	0.01
Accuracy	0.01	0.02	0.03	0.03	0.01	0.01	0.01	0.01	0.02	0.02
Fscore	0.08	0.09	0.12	0.13	0.05	0.06	0.06	0.08	0.07	0.07
Fmeasure	0.08	0.09	0.11	0.12	0.06	0.07	0.06	0.08	0.07	0.08
Rareness level =15%										
Sensitivity	0.09	0.10	0.16	0.16	0.07	0.07	0.06	0.09	0.09	0.07
Specificity	0.01	0.01	0.03	0.06	0.01	0.01	0.01	0.02	0.02	0.01
Accuracy	0.01	0.02	0.03	0.05	0.01	0.01	0.01	0.02	0.02	0.01
Fscore	0.05	0.06	0.10	0.11	0.04	0.04	0.03	0.05	0.06	0.04
Fmeasure	0.06	0.06	0.09	0.11	0.04	0.04	0.04	0.05	0.06	0.05
Rareness level =25%										
Sensitivity	0.05	0.07	0.17	0.17	0.03	0.03	0.03	0.04	0.05	0.03
Specificity	0.02	0.02	0.03	0.04	0.01	0.01	0.01	0.02	0.02	0.01
Accuracy	0.01	0.02	0.04	0.04	0.01	0.01	0.01	0.01	0.02	0.01
Fscore	0.02	0.04	0.11	0.11	0.01	0.02	0.01	0.02	0.03	0.02
Fmeasure	0.03	0.03	0.10	0.10	0.02	0.02	0.02	0.03	0.03	0.02
Rareness level =35%										
Sensitivity	0.04	0.08	0.14	0.14	0.02	0.02	0.02	0.02	0.04	0.02
Specificity	0.02	0.03	0.04	0.04	0.01	0.01	0.01	0.01	0.02	0.01
Accuracy	0.01	0.03	0.04	0.04	0.01	0.01	0.01	0.01	0.02	0.01
Fscore	0.02	0.05	0.09	0.09	0.01	0.01	0.01	0.01	0.02	0.01
Fmeasure	0.02	0.05	0.09	0.09	0.01	0.02	0.01	0.02	0.02	0.02

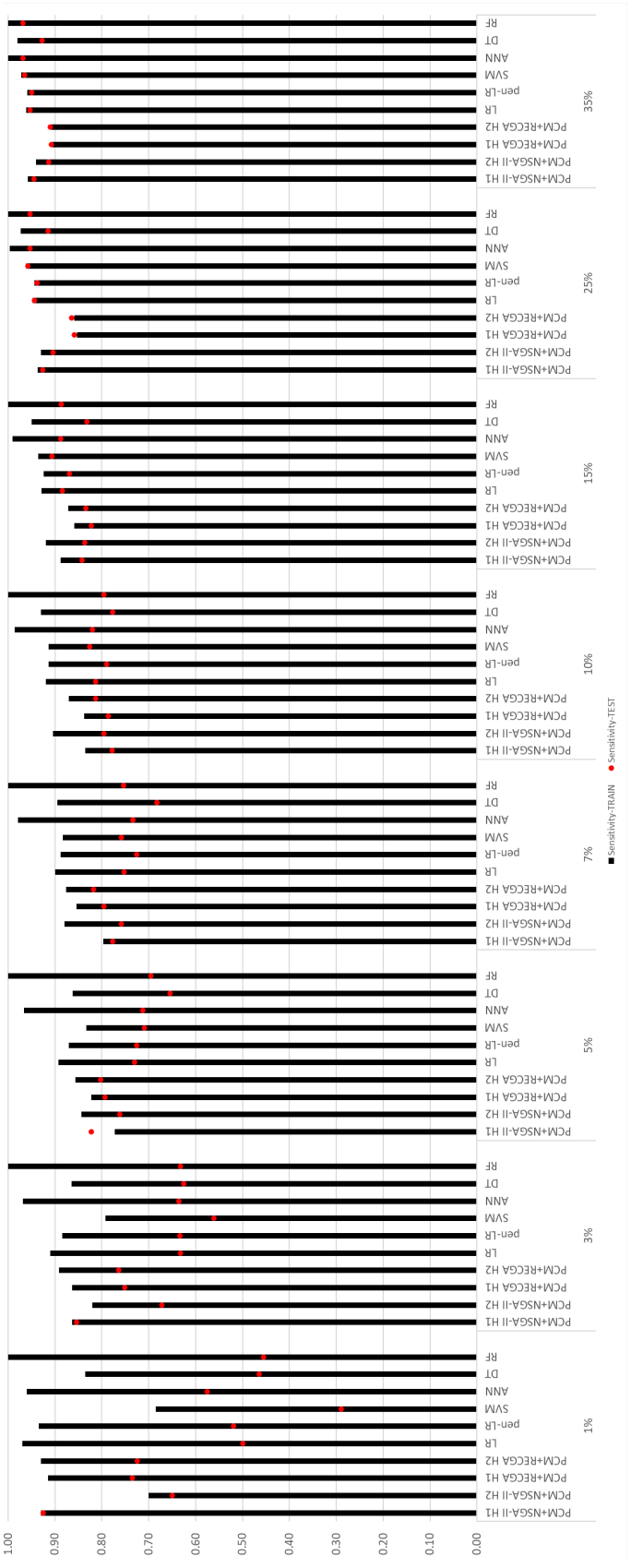


Figure 5.1: Training vs. Test Performances (WBCO Dataset)

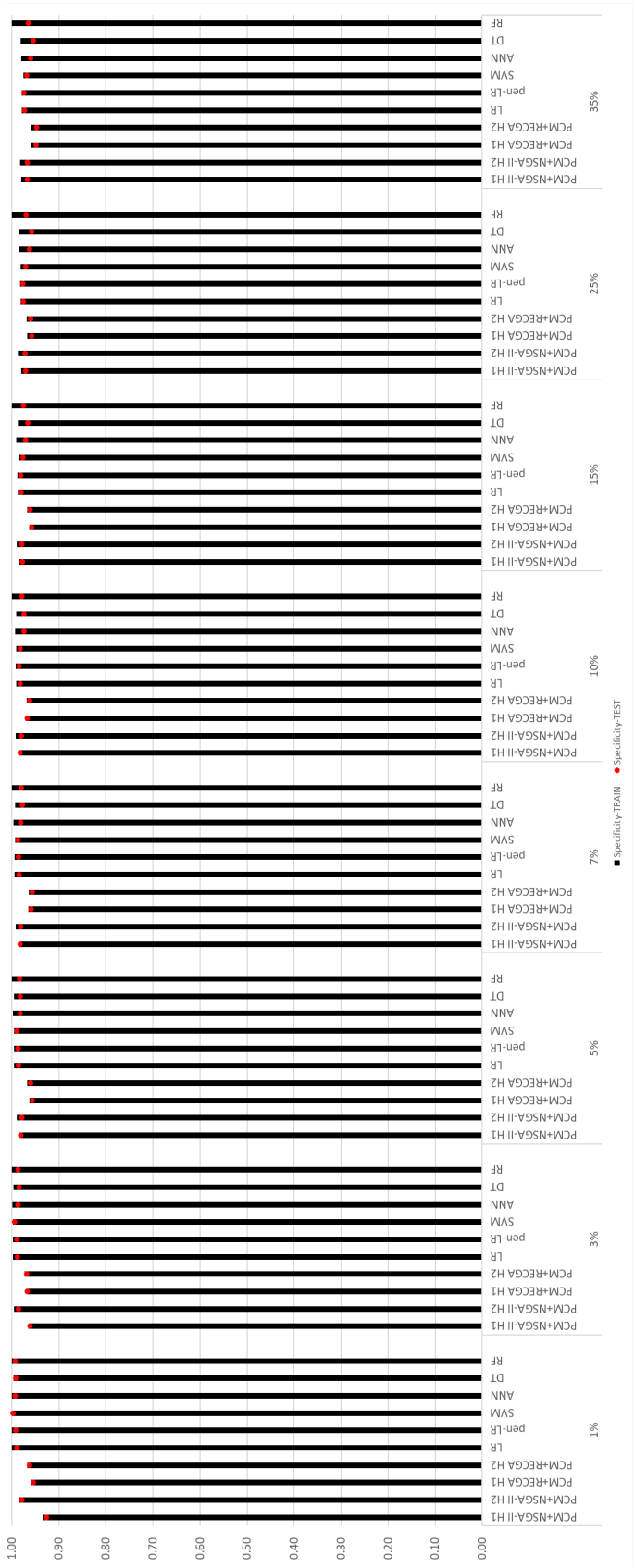


Figure 5.1: Training vs. Test Performances (WBCO Dataset)

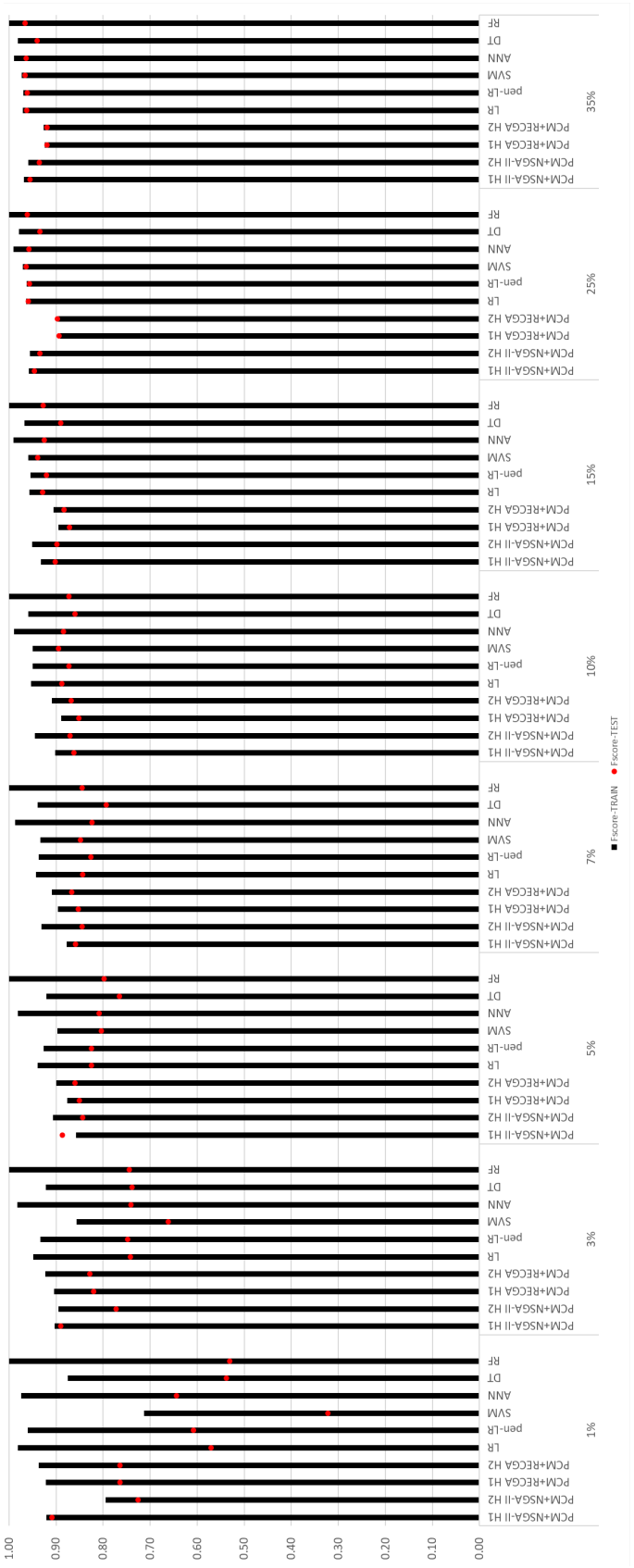


Figure 5.1: Training vs. Test Performances (WBCO Dataset)

When the model performances in training and test are observed, the experimental analysis suggest that, in terms of specificity, all models are quite successful, in all levels of rareness. However, even if some of the competitor models' training performances in sensitivity are high, their performances in test is poor and this incompatibility is explicit for low levels of rareness. For example, in training, the sensitivity performance of RF is 1.00 for all configurations, but in test, its sensitivity in 1%, 3%, 5%, 7% and 10% rareness are 0.46, 0.63, 0.70, 0.75 and 0.80, respectively. As it can be observed from Figure 5.2, for all models, the gap between training and test is very small for specificity. Additionally, as the rareness level grows, the gap diminishes for sensitivity and Fscore. The reason is that, for the configurations where the rareness level is high, there are more positive observations in the training sets. Thus, the models are able to learn the specifications of these observations more accurately.

For rareness levels less than 10% (i.e. 1%, 3%, 5% and 7%), the gap between training and test performances of PCM+NSGA-II and PCM+RECGA are always less than the average gap of the models, in terms of sensitivity and Fscore.

Therefore, it can be claimed that, while most of the competitor models tend to overfit to the training sample, PCM+NSGA-II and PCM+RECGA are much more generalizable, when one class of observations are rare compared to other.

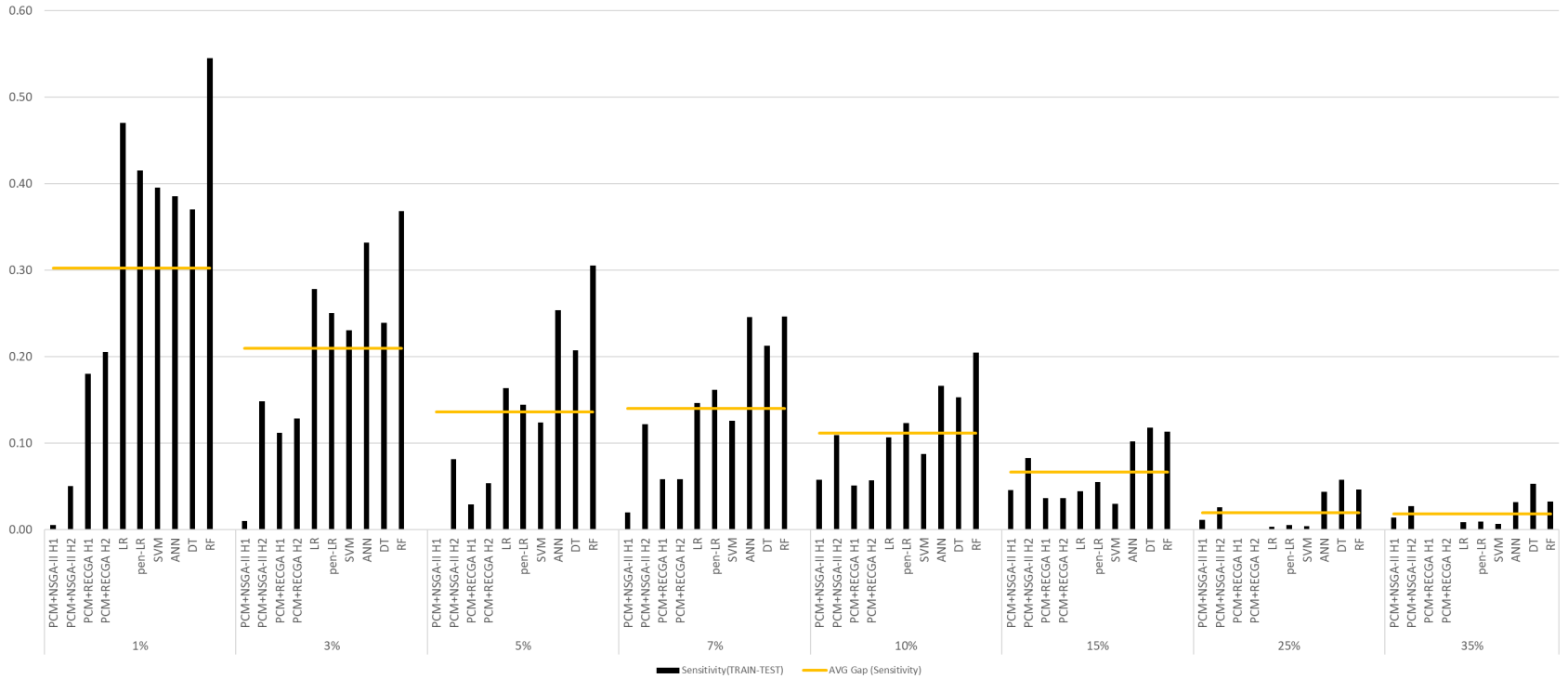


Figure 5.2: Gap Between Training and Test Performances (WBCO Dataset)

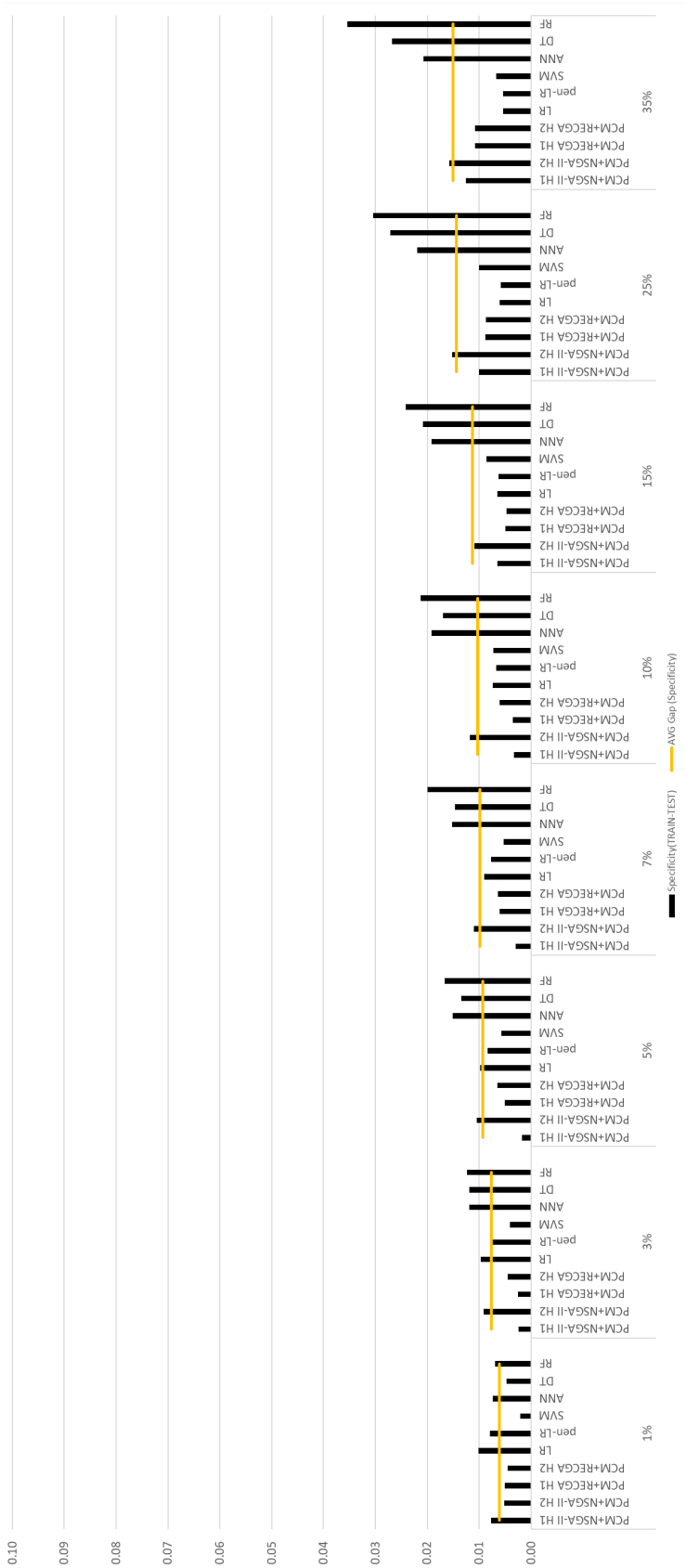


Figure 5.2: Gap Between Training and Test Performances (WBCO Dataset)

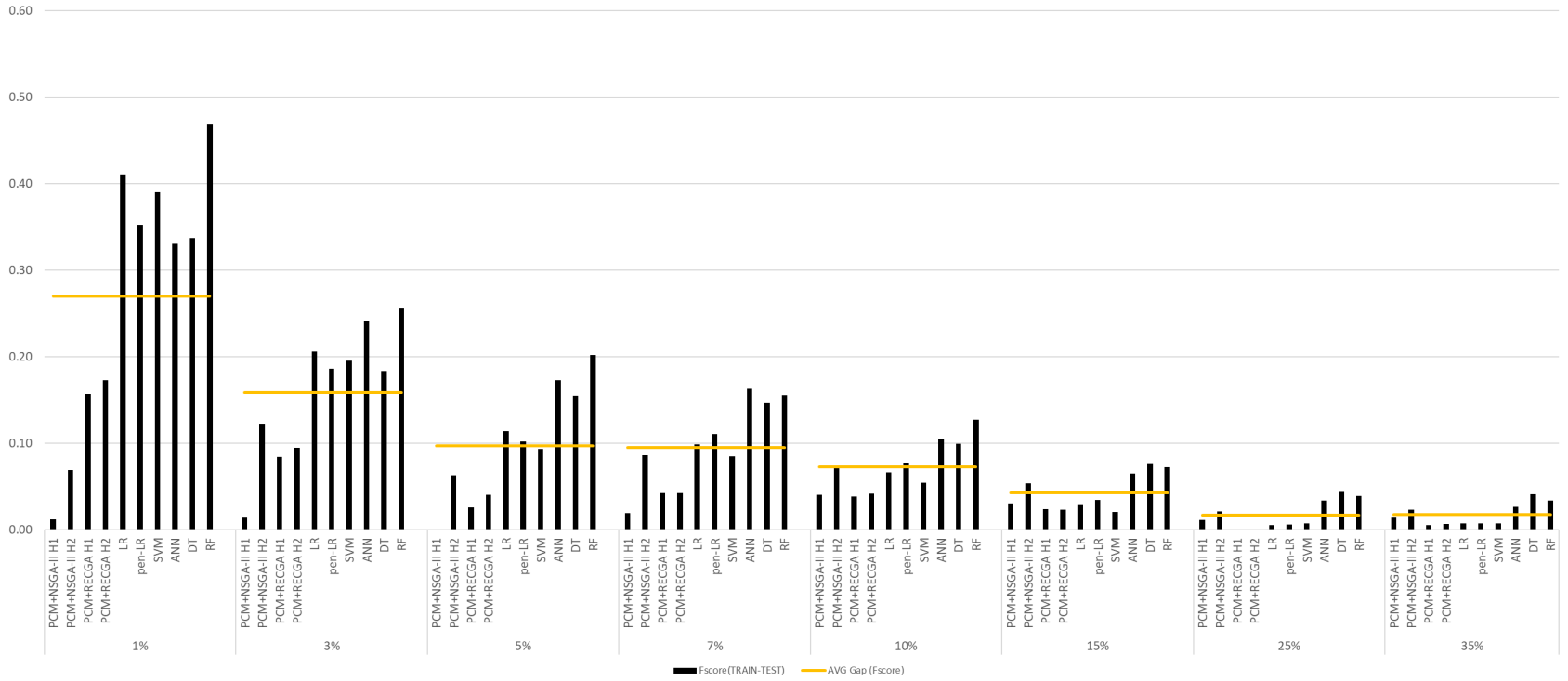


Figure 5.2: Gap Between Training and Test Performances (WBCO Dataset)

The graphics in Figure 5.3 give the average test performances and standard deviations of sensitivity, specificity and Fscore.

As it is observed from Table 5.21 and Figure 5.3, PCM+RECGA is less sensitive than PCM+NSGA-II to the changes in hyper-parameter choices. Its performance does not change significantly for hyper-parameter sets H1 and H2, at any level of rareness. However, for low levels of rarity, the hyper-parameter set represented with H1 gives strongly better results for PCM+NSGA-II. Recall that, H1 is determined with the dataset where the positive observations are extremely rare in the population. Therefore, this result is not surprising. It is also expected that, if the hyper-parameter optimization process is repeated for all levels of rarity, the model performances could be better. It is also observed that, for high levels of rareness, the hyper-parameter selection does not change model performances as much as it does in the low levels of rareness.

The graphics also indicate that, for high levels of rareness, all models perform good and regardless of the ratio of positive observations in the population, the specificity performances are high, in general. However, in terms of sensitivity and Fscore, PCM+NSGA-II and PCM+RECGA are far better than all the competitor models for 1% of rarity. The most successful model, PCM+NSGA-II^{H1}, has quite high performances with 0.93 of sensitivity, 0.93 of specificity and 0.91 of Fscore. The superiority of the suggested models proceeds for the rareness levels of 3% and 5%. They have a strong ability to discriminate positive and negative cases even the number of positive observations in training set is significantly few.

As the rarity of positive observations becomes greater than 10%, the performances of competitor models strengthen. However, we must note that, PCM+NSGA-II and PCM+RECGA are still good classifier algorithms.

The figures also show that, as the proportion of positive cases decreases in the population, standard deviations of sensitivity and Fscore of all models tend to grow. PCM+NSGA-II is one of the most robust models where the standard deviations do not change significantly for rare cases. On the other hand, especially in competitor models, significant amount of increase is observed. For rareness level of 1%, among the competitor models, the lowest standard deviations of sensitivity and Fscore are

observed in pen-LR with the value of 0.34. In the same rarity level, the corresponding values for PCM+NSGA-II^{H1} are 0.19 and 0.14; and for PCM+RECGA^{H1} they are 0.34 and 0.31, respectively. For 3% and 5% of rarity, similar results are observed. PCM+NSGA-II^{H1} and PCM+RECGA^{H1} have the lowest standard deviations, in general.

We do not give a detailed analysis about Fmeasure performances since it is not one of our performance indicators. Also, due to the facts that are explained in Section 3, Fmeasure is not a meaningful measure when one class of observations are extremely rare compared to other. However, for the configurations where the positive observations are not extremely rare, the performances of PCM+NSGA-II and PCM+RECGA compete well with the competitor models. Since all the competitor models solve an instance within a minute, we do not report their solution times in detail.

When we compare the sensitivity and Fscore performances of PCM+NSGA-II and PCM+RECGA (since the specificity values of both models are high, we do not compare their specificity values), it is observed that, for low levels of rareness (i.e. 1%, 3%, 5%) PCM+NSGA-II^{H1} outperforms PCM+RECGA^{H1} while their results are much closer for the remaining configurations. However, for 15%, 25% and 35%, PCM+NSGA-II^{H2} again has slightly better performances, compared to PCM+RECGA^{H2}.

More detailed tables that summarize the performances of suggested and compared models can be found in the Appendix (Section K). These tables also include the number of correct classifications as well as the ratio of correct classifications.

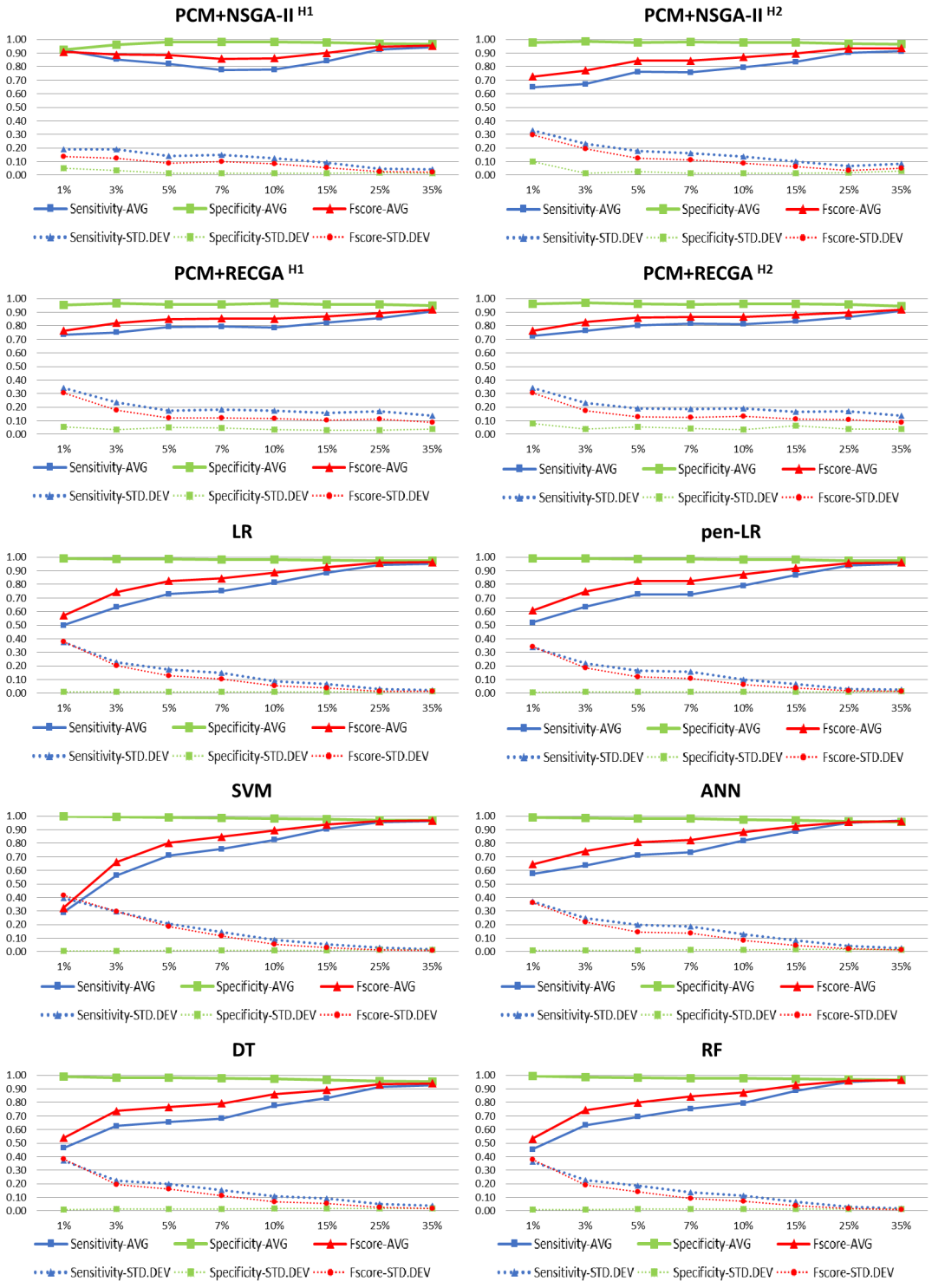


Figure 5.3: Performances for Different Rareness Levels (WBCO Dataset)

Remember that, in Section 5.1 we report the performances of the experiments in the literature which are conducted with the WBCO dataset. None of these experiments consider the rare event classification and most of them train a model with the original level of rarity of the dataset. The accuracy results range between 94-74% and 99-68%, and in most of these studies, sensitivity and specificity rates are above 96%. In the corresponding setting (when rareness level = 35%), accuracy, sensitivity and specificity performances of PCM+NSGA-II^{H2} and PCM+RECGA^{H2} are 95%, 91%, 97% and 93%, 91%, 95%, respectively. Hence, in the original rarity of the dataset, PCM+NSGA-II and PCM+RECGA can compete with the models suggested in the literature.

5.5.2 Wisconsin Breast Cancer Diagnostic Dataset

5.5.2.1 Role of PCM to Generate Initial Solutions to the Evolutionary Algorithms

As in the previous sections, we start with analyzing the performances of PCM+NSGA-II, Random+NSGA-II, PCM+RECGA and Random+RECGA, to evaluate the effect of generating initial solutions of evolutionary algorithms via the MILP model, PCM, or random. Tables 5.28, 5.29, 5.30 and 5.31 summarize the training and test performances of these models.

It is observed that, both in training and test, Random+NSGA-II has higher sensitivity values compared to PCM+NSGA-II. However, its specificity values are poor. The biased results are observed in the Fscore performances, as well. That is, PCM+NSGA-II has much better Fscore values than Random+NSGA-II (regardless of whether the hyper-parameter set is H1 or H2). For the configurations where the rareness level is low, the performance of Random+NSGA-II in specificity are quite low, but as the rareness level grows, its performance improves. However, it still lags behind the PCM+NSGA-II both in terms of specificity and Fscore.

The same analyses conducted for PCM+RECGA and Random+RECGA give similar results. Both in training and test, although the sensitivity performances of Random+RECGA is higher than that of PCM+RECGA, it has poor performance in speci-

ficity. Thus, regardless of whether the experiments are conducted with the hyper-parameter set H1 or H2, Fsore values of PCM+RECGA are almost always better than Fscore of Random+RECGA. For low levels of rareness, the failure of Random+RECGA is much more obvious. As the rareness grows it can achieve better performances, nevertheless, PCM+RECGA outperforms Random+RECGA in specificity and Fscore.

These results indicate that, in the experiments conducted with the WBCD dataset, Random+NSGA-II and Random+RECGA tend to classify most of the patients as positive, yielding poor performances in specificity and Fscore. Thus, we can conclude that it is preferable to obtain the initial solution set of evolutionary algorithms via PCM rather than to randomly generate it.

Table 5.23: Average Training Performances: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA (WBCD Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level= 1%							
Sensitivity	0.89	1.00	0.99	1.00	1.00	1.00	1.00	1.00
Specificity	0.78	0.54	0.50	0.48	0.89	0.89	0.56	0.49
Accuracy	0.78	0.54	0.51	0.48	0.89	0.89	0.56	0.49
Fscore	0.77	0.69	0.66	0.64	0.93	0.94	0.72	0.65
Fmeasure	0.08	0.04	0.03	0.03	0.37	0.39	0.04	0.03
	Rareness level=3%							
Sensitivity	0.84	0.94	0.95	0.98	0.97	0.95	1.00	1.00
Specificity	0.95	0.78	0.53	0.49	0.92	0.94	0.56	0.52
Accuracy	0.94	0.79	0.54	0.51	0.93	0.94	0.58	0.53
Fscore	0.88	0.85	0.67	0.65	0.94	0.94	0.72	0.68
Fmeasure	0.52	0.24	0.12	0.12	0.56	0.59	0.13	0.12
	Rareness level=5%							
Sensitivity	0.81	0.92	0.95	0.97	0.94	0.93	0.99	0.99
Specificity	0.96	0.82	0.56	0.51	0.93	0.94	0.57	0.53
Accuracy	0.96	0.82	0.58	0.53	0.93	0.94	0.59	0.55
Fscore	0.87	0.86	0.70	0.67	0.93	0.93	0.72	0.69
Fmeasure	0.66	0.35	0.18	0.17	0.65	0.69	0.19	0.18
	Rareness level=7%							
Sensitivity	0.76	0.90	0.95	0.97	0.91	0.89	0.98	0.99
Specificity	0.97	0.86	0.59	0.54	0.92	0.93	0.59	0.55
Accuracy	0.96	0.87	0.62	0.57	0.92	0.93	0.61	0.58
Fscore	0.85	0.88	0.73	0.69	0.91	0.90	0.73	0.71
Fmeasure	0.73	0.51	0.26	0.24	0.65	0.66	0.26	0.25
	Rareness level=10%							
Sensitivity	0.82	0.91	0.96	0.97	0.91	0.91	0.99	0.99
Specificity	0.98	0.89	0.63	0.56	0.93	0.93	0.60	0.59
Accuracy	0.96	0.89	0.66	0.60	0.93	0.93	0.64	0.63
Fscore	0.89	0.89	0.76	0.71	0.92	0.92	0.74	0.73
Fmeasure	0.81	0.63	0.36	0.33	0.73	0.75	0.35	0.35
	Rareness level=15%							
Sensitivity	0.85	0.87	0.96	0.98	0.90	0.91	0.98	0.98
Specificity	0.98	0.93	0.69	0.60	0.93	0.94	0.65	0.63
Accuracy	0.96	0.92	0.73	0.66	0.92	0.93	0.70	0.68
Fscore	0.91	0.90	0.80	0.74	0.91	0.92	0.78	0.77
Fmeasure	0.86	0.77	0.52	0.46	0.78	0.80	0.50	0.49
	Rareness level=25%							
Sensitivity	0.91	0.89	0.96	0.98	0.92	0.92	0.96	0.96
Specificity	0.97	0.94	0.77	0.68	0.93	0.93	0.72	0.70
Accuracy	0.95	0.93	0.82	0.76	0.93	0.93	0.78	0.77
Fscore	0.93	0.91	0.86	0.80	0.92	0.92	0.82	0.81
Fmeasure	0.90	0.86	0.73	0.67	0.86	0.87	0.69	0.68
	Rareness level=37%							
Sensitivity	0.94	0.92	0.97	0.98	0.95	0.94	0.96	0.96
Specificity	0.96	0.94	0.86	0.77	0.91	0.92	0.83	0.83
Accuracy	0.95	0.93	0.90	0.85	0.92	0.93	0.88	0.87
Fscore	0.95	0.93	0.91	0.86	0.93	0.93	0.89	0.89
Fmeasure	0.93	0.91	0.87	0.83	0.90	0.91	0.86	0.85

Table 5.24: Standard Deviations of Training Performance Indicators: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA (WBCD Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level=1%							
Sensitivity	0.00	0.31	0.00	0.10	0.00	0.00	0.00	0.00
Specificity	0.07	0.13	0.05	0.07	0.15	0.14	0.06	0.05
Accuracy	0.07	0.13	0.05	0.07	0.15	0.14	0.06	0.05
Fscore	0.06	0.28	0.05	0.09	0.10	0.09	0.06	0.04
Fmeasure	0.01	0.06	0.00	0.01	0.32	0.31	0.00	0.00
	Rareness level=3%							
Sensitivity	0.12	0.16	0.07	0.11	0.12	0.08	0.02	0.00
Specificity	0.08	0.04	0.05	0.06	0.06	0.08	0.06	0.05
Accuracy	0.08	0.04	0.05	0.06	0.06	0.07	0.06	0.05
Fscore	0.07	0.09	0.05	0.05	0.08	0.06	0.05	0.04
Fmeasure	0.08	0.14	0.01	0.02	0.24	0.24	0.01	0.02
	Rareness level=5%							
Sensitivity	0.10	0.14	0.07	0.09	0.10	0.09	0.05	0.04
Specificity	0.06	0.03	0.06	0.06	0.07	0.07	0.06	0.05
Accuracy	0.05	0.03	0.05	0.06	0.06	0.07	0.06	0.04
Fscore	0.06	0.09	0.05	0.05	0.06	0.05	0.05	0.04
Fmeasure	0.08	0.14	0.02	0.02	0.22	0.22	0.02	0.02
	Rareness level=7%							
Sensitivity	0.09	0.11	0.05	0.06	0.10	0.07	0.04	0.05
Specificity	0.06	0.02	0.06	0.05	0.06	0.06	0.06	0.05
Accuracy	0.06	0.02	0.06	0.05	0.06	0.06	0.05	0.04
Fscore	0.05	0.07	0.05	0.04	0.06	0.04	0.05	0.04
Fmeasure	0.10	0.10	0.03	0.03	0.14	0.15	0.02	0.02
	Rareness level=10%							
Sensitivity	0.08	0.08	0.04	0.04	0.08	0.08	0.03	0.02
Specificity	0.05	0.02	0.06	0.05	0.06	0.05	0.05	0.05
Accuracy	0.04	0.02	0.05	0.05	0.05	0.04	0.05	0.05
Fscore	0.04	0.05	0.05	0.04	0.05	0.04	0.04	0.04
Fmeasure	0.09	0.08	0.03	0.04	0.13	0.11	0.03	0.03
	Rareness level=15%							
Sensitivity	0.07	0.04	0.03	0.03	0.06	0.07	0.03	0.03
Specificity	0.03	0.02	0.06	0.05	0.04	0.03	0.05	0.05
Accuracy	0.03	0.02	0.05	0.04	0.03	0.03	0.04	0.04
Fscore	0.04	0.03	0.05	0.03	0.04	0.03	0.04	0.04
Fmeasure	0.07	0.05	0.04	0.04	0.07	0.06	0.04	0.04
	Rareness level=25%							
Sensitivity	0.04	0.03	0.02	0.02	0.04	0.05	0.02	0.02
Specificity	0.03	0.02	0.05	0.05	0.04	0.04	0.05	0.04
Accuracy	0.02	0.02	0.04	0.04	0.03	0.02	0.03	0.03
Fscore	0.02	0.02	0.04	0.03	0.03	0.02	0.03	0.03
Fmeasure	0.04	0.03	0.04	0.04	0.04	0.04	0.03	0.03
	Rareness level=37%							
Sensitivity	0.03	0.02	0.02	0.02	0.03	0.03	0.02	0.02
Specificity	0.03	0.02	0.05	0.03	0.03	0.03	0.03	0.03
Accuracy	0.02	0.02	0.03	0.02	0.02	0.01	0.02	0.02
Fscore	0.02	0.02	0.03	0.02	0.02	0.01	0.02	0.02
Fmeasure	0.02	0.02	0.03	0.02	0.02	0.02	0.02	0.03

Table 5.25: Average Test Performances: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA (WBCD Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level=1%							
Sensitivity	0.94	0.99	0.98	0.99	0.77	0.72	0.94	1.00
Specificity	0.78	0.56	0.52	0.50	0.88	0.89	0.56	0.51
Accuracy	0.79	0.56	0.52	0.50	0.88	0.89	0.56	0.52
Fscore	0.82	0.71	0.67	0.66	0.70	0.67	0.67	0.68
Fmeasure	0.10	0.04	0.03	0.03	0.24	0.26	0.04	0.03
	Rareness level=3%							
Sensitivity	0.79	0.92	0.95	0.98	0.76	0.75	0.97	1.00
Specificity	0.95	0.81	0.54	0.52	0.92	0.94	0.57	0.54
Accuracy	0.95	0.81	0.56	0.53	0.92	0.93	0.58	0.56
Fscore	0.85	0.85	0.69	0.67	0.80	0.80	0.71	0.70
Fmeasure	0.54	0.26	0.12	0.12	0.43	0.46	0.13	0.13
	Rareness level=5%							
Sensitivity	0.77	0.94	0.95	0.99	0.80	0.77	0.99	1.00
Specificity	0.97	0.84	0.57	0.54	0.93	0.94	0.57	0.55
Accuracy	0.96	0.85	0.59	0.56	0.92	0.93	0.59	0.57
Fscore	0.84	0.88	0.71	0.69	0.84	0.82	0.72	0.71
Fmeasure	0.65	0.39	0.18	0.18	0.56	0.57	0.19	0.19
	Rareness level=7%							
Sensitivity	0.76	0.92	0.95	0.99	0.85	0.84	0.98	0.99
Specificity	0.98	0.87	0.60	0.56	0.92	0.93	0.59	0.58
Accuracy	0.96	0.88	0.62	0.59	0.92	0.92	0.62	0.61
Fscore	0.84	0.89	0.73	0.71	0.87	0.87	0.74	0.73
Fmeasure	0.73	0.54	0.26	0.25	0.62	0.63	0.27	0.26
	Rareness level=10%							
Sensitivity	0.75	0.89	0.93	0.97	0.83	0.82	0.98	0.98
Specificity	0.98	0.90	0.63	0.59	0.93	0.93	0.61	0.61
Accuracy	0.95	0.90	0.66	0.63	0.92	0.92	0.65	0.65
Fscore	0.84	0.89	0.75	0.73	0.87	0.87	0.75	0.75
Fmeasure	0.76	0.65	0.35	0.34	0.68	0.69	0.35	0.35
	Rareness level=15%							
Sensitivity	0.81	0.87	0.96	0.98	0.87	0.87	0.98	0.98
Specificity	0.97	0.93	0.69	0.63	0.93	0.94	0.67	0.66
Accuracy	0.95	0.92	0.73	0.69	0.92	0.93	0.71	0.71
Fscore	0.88	0.90	0.80	0.77	0.90	0.90	0.79	0.79
Fmeasure	0.83	0.78	0.52	0.49	0.77	0.78	0.51	0.50
	Rareness level=25%							
Sensitivity	0.88	0.88	0.97	0.99	0.91	0.90	0.98	0.98
Specificity	0.96	0.94	0.78	0.70	0.93	0.93	0.73	0.73
Accuracy	0.94	0.93	0.83	0.77	0.92	0.93	0.79	0.79
Fscore	0.92	0.91	0.86	0.82	0.92	0.92	0.83	0.83
Fmeasure	0.88	0.86	0.74	0.69	0.86	0.86	0.70	0.70
	Rareness level=37%							
Sensitivity	0.92	0.91	0.95	0.97	0.94	0.93	0.95	0.95
Specificity	0.95	0.95	0.85	0.79	0.91	0.93	0.84	0.84
Accuracy	0.94	0.93	0.89	0.86	0.92	0.93	0.88	0.88
Fscore	0.93	0.93	0.90	0.87	0.93	0.93	0.89	0.89
Fmeasure	0.92	0.91	0.87	0.83	0.90	0.91	0.86	0.85

Table 5.26: Standard Deviations of Test Performance Indicators: PCM+NSGA-II vs. Random+NSGA-II and PCM+RECGA vs. Random+RECGA (WBCD Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level=1%							
Sensitivity	0.10	0.24	0.10	0.14	0.45	0.42	0.00	0.24
Specificity	0.08	0.13	0.05	0.06	0.15	0.14	0.06	0.07
Accuracy	0.08	0.13	0.05	0.06	0.15	0.14	0.06	0.06
Fscore	0.10	0.22	0.08	0.11	0.42	0.39	0.05	0.18
Fmeasure	0.01	0.10	0.00	0.01	0.31	0.28	0.00	0.01
	Rareness level=3%							
Sensitivity	0.13	0.19	0.06	0.11	0.25	0.24	0.04	0.10
Specificity	0.07	0.04	0.05	0.06	0.05	0.08	0.05	0.06
Accuracy	0.07	0.04	0.05	0.06	0.05	0.07	0.05	0.06
Fscore	0.07	0.13	0.05	0.06	0.19	0.17	0.05	0.05
Fmeasure	0.08	0.18	0.01	0.02	0.19	0.17	0.01	0.02
	Rareness level=5%							
Sensitivity	0.10	0.21	0.05	0.10	0.22	0.21	0.02	0.05
Specificity	0.06	0.02	0.05	0.06	0.06	0.07	0.06	0.06
Accuracy	0.06	0.02	0.05	0.05	0.06	0.07	0.05	0.06
Fscore	0.06	0.15	0.05	0.06	0.16	0.15	0.05	0.05
Fmeasure	0.09	0.14	0.02	0.03	0.20	0.19	0.02	0.02
	Rareness level=7%							
Sensitivity	0.09	0.17	0.04	0.09	0.14	0.15	0.04	0.06
Specificity	0.07	0.03	0.06	0.06	0.06	0.06	0.05	0.05
Accuracy	0.06	0.02	0.05	0.05	0.05	0.05	0.05	0.05
Fscore	0.05	0.12	0.05	0.05	0.08	0.09	0.04	0.04
Fmeasure	0.12	0.12	0.03	0.04	0.14	0.14	0.03	0.03
	Rareness level=10%							
Sensitivity	0.10	0.13	0.05	0.07	0.13	0.13	0.04	0.04
Specificity	0.05	0.02	0.05	0.06	0.05	0.05	0.05	0.05
Accuracy	0.04	0.02	0.05	0.05	0.05	0.04	0.04	0.05
Fscore	0.06	0.08	0.04	0.04	0.07	0.07	0.04	0.04
Fmeasure	0.10	0.10	0.03	0.04	0.12	0.11	0.03	0.03
	Rareness level=15%							
Sensitivity	0.10	0.10	0.03	0.05	0.10	0.10	0.03	0.03
Specificity	0.04	0.02	0.05	0.05	0.04	0.03	0.05	0.05
Accuracy	0.03	0.02	0.04	0.04	0.03	0.03	0.04	0.04
Fscore	0.05	0.06	0.04	0.04	0.05	0.05	0.03	0.04
Fmeasure	0.08	0.07	0.03	0.05	0.08	0.07	0.03	0.04
	Rareness level=25%							
Sensitivity	0.06	0.05	0.02	0.03	0.05	0.06	0.02	0.02
Specificity	0.03	0.02	0.04	0.05	0.04	0.04	0.04	0.04
Accuracy	0.02	0.02	0.03	0.04	0.03	0.03	0.03	0.03
Fscore	0.03	0.03	0.03	0.03	0.03	0.03	0.02	0.02
Fmeasure	0.04	0.04	0.03	0.04	0.05	0.04	0.03	0.03
	Rareness level=37%							
Sensitivity	0.04	0.04	0.02	0.02	0.04	0.03	0.02	0.02
Specificity	0.03	0.03	0.04	0.04	0.04	0.03	0.03	0.03
Accuracy	0.02	0.02	0.03	0.03	0.03	0.02	0.02	0.02
Fscore	0.02	0.02	0.03	0.02	0.03	0.02	0.02	0.02
Fmeasure	0.03	0.03	0.03	0.03	0.03	0.03	0.02	0.02

Table 5.27 summarizes the solution times of PCM+NSGA-II, Random+NSGA-II, PCM+RECGA and Random+RECGA.

Table 5.27: Solution Times (in sec.) (WBCD Dataset)

	PCM+NSGA-II		Random+NSGA-II		PCM+RECGA		Random+RECGA	
	H1	H2	H1	H2	H1	H2	H1	H2
	Rareness level=1%							
AVG	59.64	84.31	12.22	38.24	86.47	89.73	50.76	16.58
STD.DEV	7.07	1.75	1.53	0.45	22.61	13.44	0.41	8.40
	Rareness level=3%							
AVG	61.97	113.70	13.10	40.36	96.40	90.01	52.06	17.68
STD.DEV	6.29	0.77	1.51	0.47	10.72	34.06	1.07	8.01
	Rareness level=5%							
AVG	63.48	89.07	13.65	41.29	89.29	91.25	52.55	18.72
STD.DEV	1.39	1.00	1.41	0.40	4.23	16.15	1.73	8.29
	Rareness level=7%							
AVG	66.34	91.93	14.23	43.40	92.60	93.47	54.81	18.67
STD.DEV	1.53	6.71	1.36	0.36	5.28	15.75	0.40	7.91
	Rareness level=10%							
AVG	67.00	96.44	15.13	46.29	95.87	99.32	56.06	19.81
STD.DEV	2.14	1.68	1.37	0.37	5.66	15.90	0.46	6.06
	Rareness level=15%							
AVG	74.46	107.15	16.89	53.04	102.98	108.84	62.51	22.25
STD.DEV	2.81	2.49	1.71	0.39	6.95	18.66	0.45	7.57
	Rareness level=25%							
AVG	103.45	142.66	21.45	70.18	132.02	150.12	79.98	27.85
STD.DEV	8.17	10.94	1.77	0.42	10.13	23.40	0.43	6.79
	Rareness level=37%							
AVG	144.35	203.53	30.00	104.17	180.83	209.12	113.11	39.31
STD.DEV	13.88	10.04	5.56	0.59	15.49	26.60	0.55	10.69

5.5.2.2 Comparison of PCM+NSGA-II, PCM+RECGA and Competitor Models

In this section, to compare PCM+NSGA-II, PCM+RECGA and the competitor models, we summarize the average performances and standard deviations of performance indicators in Tables 5.28, 5.29, 5.30 and 5.31 for training and test, respectively.

Based on the results given in Tables 5.28 and 5.30, Figure 5.4 shows the training and test performances of the models and Figure 5.5 illustrates the gaps between the models' training and test performances. Note that, the gap refers to how much the training performance is greater than the test performance and if the training performance lags behind the test, the gap is expressed as zero.

Table 5.28: Average Training Performances (WBCD Dataset)

	PCM+NSGA-II^{H1}	PCM+NSGA-II^{H2}	PCM+RECGA^{H1}	PCM+RECGA^{H2}	LR	pen-LR	SVM	ANN	DT	RF
	Rareness level=1%									
Sensitivity	0.89	1.00	1.00	1.00	1.00	0.95	0.64	1.00	0.86	1.00
Specificity	0.78	0.54	0.89	0.89	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	0.78	0.54	0.89	0.89	1.00	1.00	1.00	1.00	1.00	1.00
Fscore	0.77	0.69	0.93	0.94	1.00	0.96	0.64	1.00	0.90	1.00
Fmeasure	0.08	0.04	0.37	0.39	1.00	0.96	0.64	1.00	0.87	1.00
	Rareness level=3%									
Sensitivity	0.84	0.94	0.97	0.95	1.00	0.97	0.89	1.00	0.90	1.00
Specificity	0.95	0.78	0.92	0.94	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	0.94	0.79	0.93	0.94	1.00	1.00	1.00	1.00	0.99	1.00
Fscore	0.88	0.85	0.94	0.94	1.00	0.98	0.92	1.00	0.94	1.00
Fmeasure	0.52	0.24	0.56	0.59	1.00	0.98	0.92	1.00	0.91	1.00
	Rareness level=5%									
Sensitivity	0.81	0.92	0.94	0.93	0.99	0.97	0.90	1.00	0.89	1.00
Specificity	0.96	0.82	0.93	0.94	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	0.96	0.82	0.93	0.94	1.00	1.00	1.00	1.00	0.99	1.00
Fscore	0.87	0.86	0.93	0.93	1.00	0.98	0.93	1.00	0.94	1.00
Fmeasure	0.66	0.35	0.65	0.69	1.00	0.98	0.93	1.00	0.93	1.00
	Rareness level=7%									
Sensitivity	0.76	0.90	0.91	0.89	0.99	0.96	0.89	1.00	0.90	1.00
Specificity	0.97	0.86	0.92	0.93	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	0.96	0.87	0.92	0.93	1.00	1.00	0.99	1.00	0.99	1.00
Fscore	0.85	0.88	0.91	0.90	1.00	0.98	0.93	1.00	0.95	1.00
Fmeasure	0.73	0.51	0.65	0.66	1.00	0.98	0.93	1.00	0.93	1.00
	Rareness level=10%									
Sensitivity	0.82	0.91	0.91	0.91	0.98	0.95	0.92	1.00	0.91	1.00
Specificity	0.98	0.89	0.93	0.93	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	0.96	0.89	0.93	0.93	1.00	1.00	0.99	1.00	0.99	1.00
Fscore	0.89	0.89	0.92	0.92	0.99	0.98	0.95	1.00	0.95	1.00
Fmeasure	0.81	0.63	0.73	0.75	0.99	0.97	0.95	1.00	0.94	1.00
	Rareness level=15%									
Sensitivity	0.85	0.87	0.90	0.91	0.98	0.96	0.94	1.00	0.94	1.00
Specificity	0.98	0.93	0.93	0.94	1.00	1.00	1.00	1.00	0.99	1.00
Accuracy	0.96	0.92	0.92	0.93	1.00	0.99	0.99	1.00	0.99	1.00
Fscore	0.91	0.90	0.91	0.92	0.99	0.98	0.97	1.00	0.97	1.00
Fmeasure	0.86	0.77	0.78	0.80	0.99	0.98	0.97	1.00	0.95	1.00
	Rareness level=25%									
Sensitivity	0.91	0.89	0.92	0.92	0.96	0.96	0.95	0.99	0.97	1.00
Specificity	0.97	0.94	0.93	0.93	0.99	1.00	1.00	1.00	0.99	1.00
Accuracy	0.95	0.93	0.93	0.93	0.99	0.99	0.99	1.00	0.98	1.00
Fscore	0.93	0.91	0.92	0.92	0.98	0.98	0.97	1.00	0.98	1.00
Fmeasure	0.90	0.86	0.86	0.87	0.97	0.97	0.97	1.00	0.97	1.00
	Rareness level=37%									
Sensitivity	0.94	0.92	0.95	0.94	0.97	0.97	0.97	0.99	0.98	1.00
Specificity	0.96	0.94	0.91	0.92	0.99	0.99	1.00	1.00	0.99	1.00
Accuracy	0.95	0.93	0.92	0.93	0.98	0.98	0.98	0.99	0.98	1.00
Fscore	0.95	0.93	0.93	0.93	0.98	0.98	0.98	0.99	0.98	1.00
Fmeasure	0.93	0.91	0.90	0.91	0.98	0.98	0.98	0.99	0.98	1.00

Table 5.29: Standard Deviations of Training Performance Indicators (WBCD Dataset)

	PCM+NSGA-II^{H1}	PCM+NSGA-II^{H2}	PCM+RECGA^{H1}	PCM+RECGA^{H2}	LR	pen-LR	SVM	ANN	DT	RF
	Rareness level=1%									
Sensitivity	0.00	0.31	0.00	0.00	0.00	0.17	0.47	0.00	0.24	0.00
Specificity	0.07	0.13	0.15	0.14	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.07	0.13	0.15	0.14	0.00	0.00	0.00	0.00	0.00	0.00
Fscore	0.06	0.28	0.10	0.09	0.00	0.13	0.47	0.00	0.17	0.00
Fmeasure	0.01	0.06	0.32	0.31	0.00	0.13	0.47	0.00	0.17	0.00
	Rareness level=3%									
Sensitivity	0.12	0.16	0.12	0.08	0.00	0.06	0.19	0.00	0.12	0.00
Specificity	0.08	0.04	0.06	0.08	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.08	0.04	0.06	0.07	0.00	0.00	0.01	0.00	0.00	0.00
Fscore	0.07	0.09	0.08	0.06	0.00	0.03	0.17	0.00	0.07	0.00
Fmeasure	0.08	0.14	0.24	0.24	0.00	0.03	0.17	0.00	0.08	0.00
	Rareness level=5%									
Sensitivity	0.10	0.14	0.10	0.09	0.04	0.06	0.18	0.01	0.09	0.00
Specificity	0.06	0.03	0.07	0.07	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.05	0.03	0.06	0.07	0.00	0.00	0.01	0.00	0.01	0.00
Fscore	0.06	0.09	0.06	0.05	0.02	0.03	0.16	0.00	0.06	0.00
Fmeasure	0.08	0.14	0.22	0.22	0.03	0.03	0.16	0.01	0.06	0.00
	Rareness level=7%									
Sensitivity	0.09	0.11	0.10	0.07	0.02	0.04	0.15	0.00	0.07	0.00
Specificity	0.06	0.02	0.06	0.06	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.06	0.02	0.06	0.06	0.00	0.00	0.01	0.00	0.00	0.00
Fscore	0.05	0.07	0.06	0.04	0.01	0.02	0.14	0.00	0.04	0.00
Fmeasure	0.10	0.10	0.14	0.15	0.02	0.03	0.14	0.00	0.04	0.00
	Rareness level=10%									
Sensitivity	0.08	0.08	0.08	0.08	0.03	0.04	0.11	0.00	0.07	0.00
Specificity	0.05	0.02	0.06	0.05	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.04	0.02	0.05	0.04	0.00	0.00	0.01	0.00	0.01	0.00
Fscore	0.04	0.05	0.05	0.04	0.02	0.02	0.10	0.00	0.04	0.00
Fmeasure	0.09	0.08	0.13	0.11	0.02	0.02	0.10	0.00	0.04	0.00
	Rareness level=15%									
Sensitivity	0.07	0.04	0.06	0.07	0.03	0.03	0.04	0.02	0.04	0.00
Specificity	0.03	0.02	0.04	0.03	0.00	0.00	0.00	0.00	0.01	0.00
Accuracy	0.03	0.02	0.03	0.03	0.01	0.01	0.01	0.00	0.01	0.00
Fscore	0.04	0.03	0.04	0.03	0.02	0.02	0.02	0.01	0.02	0.00
Fmeasure	0.07	0.05	0.07	0.06	0.02	0.02	0.02	0.01	0.02	0.00
	Rareness level=25%									
Sensitivity	0.04	0.03	0.04	0.05	0.02	0.02	0.02	0.01	0.02	0.00
Specificity	0.03	0.02	0.04	0.04	0.00	0.00	0.00	0.00	0.01	0.00
Accuracy	0.02	0.02	0.03	0.02	0.01	0.01	0.01	0.00	0.01	0.00
Fscore	0.02	0.02	0.03	0.02	0.01	0.01	0.01	0.01	0.01	0.00
Fmeasure	0.04	0.03	0.04	0.04	0.01	0.01	0.01	0.01	0.01	0.00
	Rareness level=37%									
Sensitivity	0.03	0.02	0.03	0.03	0.01	0.01	0.01	0.01	0.01	0.00
Specificity	0.03	0.02	0.03	0.03	0.00	0.00	0.00	0.00	0.01	0.00
Accuracy	0.02	0.02	0.02	0.01	0.00	0.00	0.00	0.01	0.01	0.00
Fscore	0.02	0.02	0.02	0.01	0.01	0.00	0.01	0.01	0.01	0.00
Fmeasure	0.02	0.02	0.02	0.02	0.01	0.01	0.01	0.01	0.01	0.00

Table 5.30: Average Test Performances (WBCD Dataset)

	PCM+NSGA-II^{H1}	PCM+NSGA-II^{H2}	PCM+RECGA^{H1}	PCM+RECGA^{H2}	LR	pen-LR	SVM	ANN	DT	RF
Rareness level=1%										
Sensitivity	0.94	0.99	0.77	0.72	0.76	0.59	0.32	0.76	0.55	0.53
Specificity	0.78	0.56	0.88	0.89	1.00	0.99	1.00	1.00	1.00	1.00
Accuracy	0.79	0.56	0.88	0.89	1.00	0.99	0.99	0.99	0.99	0.99
Fscore	0.82	0.71	0.70	0.67	0.76	0.59	0.32	0.76	0.55	0.53
Fmeasure	0.10	0.04	0.24	0.26	0.70	0.49	0.32	0.69	0.50	0.47
Rareness level=3%										
Sensitivity	0.79	0.92	0.76	0.75	0.72	0.66	0.66	0.77	0.58	0.56
Specificity	0.95	0.81	0.92	0.94	0.99	0.99	1.00	0.99	0.99	0.99
Accuracy	0.95	0.81	0.92	0.93	0.99	0.98	0.99	0.99	0.98	0.98
Fscore	0.85	0.85	0.80	0.80	0.81	0.76	0.75	0.84	0.69	0.67
Fmeasure	0.54	0.26	0.43	0.46	0.74	0.69	0.72	0.79	0.61	0.59
Rareness level=5%										
Sensitivity	0.77	0.94	0.80	0.77	0.84	0.77	0.75	0.84	0.65	0.70
Specificity	0.97	0.84	0.93	0.94	0.99	0.99	1.00	0.99	0.99	0.99
Accuracy	0.96	0.85	0.92	0.93	0.98	0.98	0.98	0.98	0.97	0.97
Fscore	0.84	0.88	0.84	0.82	0.90	0.86	0.83	0.90	0.76	0.80
Fmeasure	0.65	0.39	0.56	0.57	0.84	0.81	0.80	0.84	0.69	0.71
Rareness level=7%										
Sensitivity	0.76	0.92	0.85	0.84	0.89	0.87	0.81	0.90	0.72	0.73
Specificity	0.98	0.87	0.92	0.93	0.99	0.99	1.00	0.99	0.98	0.98
Accuracy	0.96	0.88	0.92	0.92	0.98	0.98	0.98	0.98	0.96	0.97
Fscore	0.84	0.89	0.87	0.87	0.93	0.92	0.88	0.94	0.82	0.83
Fmeasure	0.73	0.54	0.62	0.63	0.86	0.87	0.86	0.87	0.74	0.75
Rareness level=10%										
Sensitivity	0.75	0.89	0.83	0.82	0.89	0.87	0.82	0.89	0.71	0.74
Specificity	0.98	0.90	0.93	0.93	0.98	0.99	0.99	0.98	0.98	0.98
Accuracy	0.95	0.90	0.92	0.92	0.97	0.98	0.98	0.98	0.95	0.96
Fscore	0.84	0.89	0.87	0.87	0.93	0.92	0.89	0.93	0.81	0.84
Fmeasure	0.76	0.65	0.68	0.69	0.87	0.89	0.87	0.88	0.75	0.76
Rareness level=15%										
Sensitivity	0.81	0.87	0.87	0.87	0.91	0.90	0.88	0.90	0.79	0.81
Specificity	0.97	0.93	0.93	0.94	0.98	0.99	0.99	0.98	0.97	0.97
Accuracy	0.95	0.92	0.92	0.93	0.97	0.97	0.98	0.97	0.94	0.95
Fscore	0.88	0.90	0.90	0.90	0.94	0.94	0.93	0.94	0.87	0.88
Fmeasure	0.83	0.78	0.77	0.78	0.90	0.91	0.92	0.89	0.80	0.83
Rareness level=25%										
Sensitivity	0.88	0.88	0.91	0.90	0.94	0.94	0.93	0.93	0.86	0.88
Specificity	0.96	0.94	0.93	0.93	0.98	0.98	0.99	0.97	0.95	0.97
Accuracy	0.94	0.93	0.92	0.93	0.97	0.97	0.98	0.96	0.93	0.95
Fscore	0.92	0.91	0.92	0.92	0.96	0.96	0.96	0.94	0.91	0.93
Fmeasure	0.88	0.86	0.86	0.86	0.94	0.94	0.95	0.92	0.86	0.90
Rareness level=37%										
Sensitivity	0.92	0.91	0.94	0.93	0.94	0.94	0.94	0.93	0.91	0.93
Specificity	0.95	0.95	0.91	0.93	0.98	0.98	0.99	0.96	0.94	0.97
Accuracy	0.94	0.93	0.92	0.93	0.96	0.96	0.97	0.95	0.93	0.96
Fscore	0.93	0.93	0.93	0.93	0.96	0.96	0.96	0.95	0.93	0.95
Fmeasure	0.92	0.91	0.90	0.91	0.95	0.95	0.95	0.93	0.91	0.94

Table 5.31: Standard Deviations of Test Performance Indicators (WBCD Dataset)

	PCM+NSGA-II^{H1}	PCM+NSGA-II^{H2}	PCM+RECGA^{H1}	PCM+RECGA^{H2}	LR	pen-LR	SVM	ANN	DT	RF
Rareness level=1%										
Sensitivity	0.10	0.24	0.45	0.42	0.43	0.49	0.47	0.43	0.50	0.50
Specificity	0.08	0.13	0.15	0.14	0.01	0.01	0.00	0.01	0.01	0.01
Accuracy	0.08	0.13	0.15	0.14	0.01	0.01	0.00	0.01	0.01	0.01
Fscore	0.10	0.22	0.42	0.39	0.43	0.49	0.47	0.43	0.50	0.50
Fmeasure	0.01	0.10	0.31	0.28	0.42	0.44	0.46	0.42	0.47	0.47
Rareness level=3%										
Sensitivity	0.13	0.19	0.25	0.24	0.23	0.27	0.28	0.23	0.26	0.25
Specificity	0.07	0.04	0.05	0.08	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.07	0.04	0.05	0.07	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.07	0.13	0.19	0.17	0.18	0.22	0.26	0.16	0.25	0.24
Fmeasure	0.08	0.18	0.19	0.17	0.18	0.22	0.25	0.18	0.24	0.24
Rareness level=5%										
Sensitivity	0.10	0.21	0.22	0.21	0.15	0.19	0.23	0.15	0.22	0.21
Specificity	0.06	0.02	0.06	0.07	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.06	0.02	0.06	0.07	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.06	0.15	0.16	0.15	0.09	0.13	0.20	0.09	0.18	0.16
Fmeasure	0.09	0.14	0.20	0.19	0.11	0.15	0.20	0.12	0.17	0.16
Rareness level=7%										
Sensitivity	0.09	0.17	0.14	0.15	0.11	0.12	0.18	0.10	0.17	0.16
Specificity	0.07	0.03	0.06	0.06	0.01	0.01	0.01	0.01	0.02	0.01
Accuracy	0.06	0.02	0.05	0.05	0.02	0.01	0.01	0.01	0.02	0.01
Fscore	0.05	0.12	0.08	0.09	0.07	0.07	0.16	0.06	0.13	0.11
Fmeasure	0.12	0.12	0.14	0.14	0.10	0.10	0.16	0.09	0.13	0.11
Rareness level=10%										
Sensitivity	0.10	0.13	0.13	0.13	0.09	0.11	0.15	0.10	0.15	0.13
Specificity	0.05	0.02	0.05	0.05	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.04	0.02	0.05	0.04	0.01	0.02	0.02	0.01	0.02	0.02
Fscore	0.06	0.08	0.07	0.07	0.05	0.07	0.12	0.06	0.10	0.09
Fmeasure	0.10	0.10	0.12	0.11	0.07	0.08	0.12	0.07	0.10	0.10
Rareness level=15%										
Sensitivity	0.10	0.10	0.10	0.10	0.07	0.08	0.09	0.07	0.11	0.09
Specificity	0.04	0.02	0.04	0.03	0.01	0.01	0.01	0.02	0.02	0.02
Accuracy	0.03	0.02	0.03	0.03	0.02	0.01	0.01	0.02	0.02	0.02
Fscore	0.05	0.06	0.05	0.05	0.04	0.04	0.06	0.04	0.07	0.06
Fmeasure	0.08	0.07	0.08	0.07	0.05	0.05	0.06	0.05	0.07	0.07
Rareness level=25%										
Sensitivity	0.06	0.05	0.05	0.06	0.04	0.04	0.04	0.05	0.06	0.05
Specificity	0.03	0.02	0.04	0.04	0.02	0.01	0.01	0.02	0.03	0.02
Accuracy	0.02	0.02	0.03	0.03	0.02	0.01	0.01	0.02	0.02	0.02
Fscore	0.03	0.03	0.03	0.03	0.02	0.02	0.02	0.03	0.03	0.03
Fmeasure	0.04	0.04	0.05	0.04	0.03	0.03	0.03	0.03	0.04	0.04
Rareness level=37%										
Sensitivity	0.04	0.04	0.04	0.03	0.03	0.03	0.06	0.03	0.04	0.03
Specificity	0.03	0.03	0.04	0.03	0.02	0.02	0.01	0.02	0.02	0.01
Accuracy	0.02	0.02	0.03	0.02	0.01	0.01	0.02	0.02	0.02	0.01
Fscore	0.02	0.02	0.03	0.02	0.01	0.02	0.05	0.02	0.02	0.02
Fmeasure	0.03	0.03	0.03	0.03	0.02	0.02	0.05	0.02	0.03	0.02

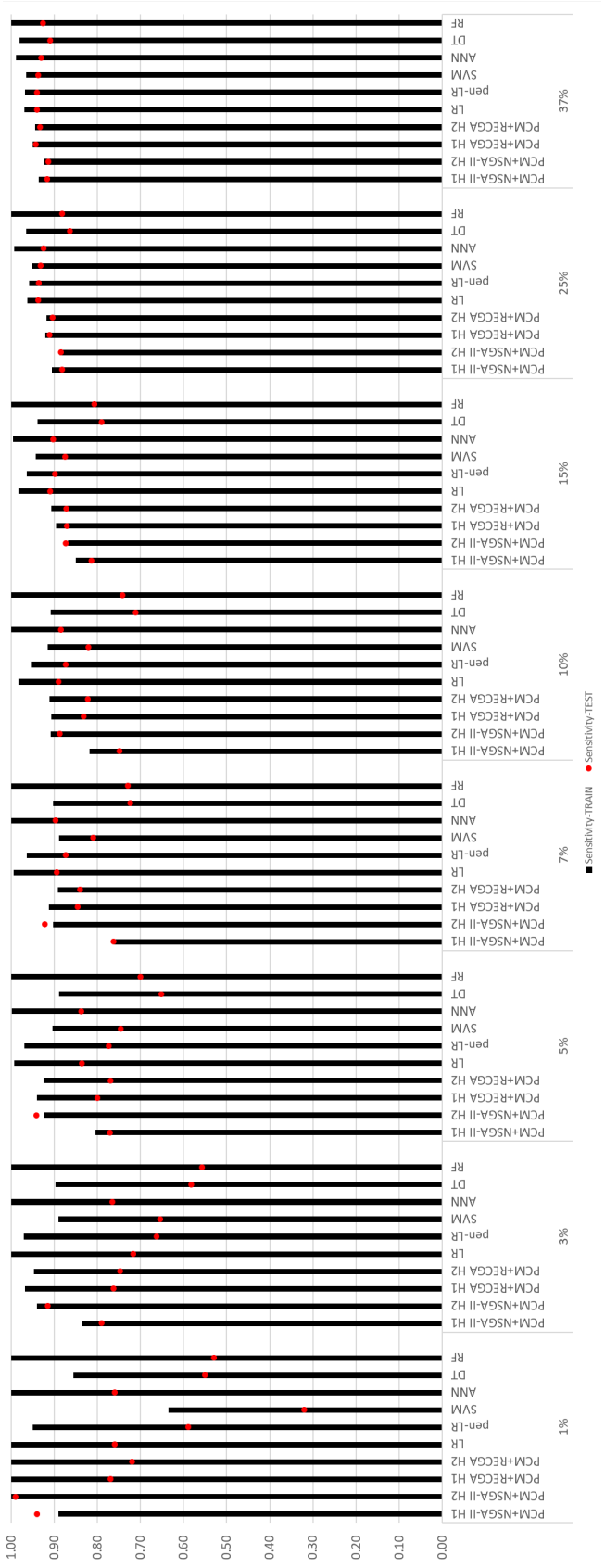


Figure 5.4: Training vs. Test Performances (WBCD Dataset)

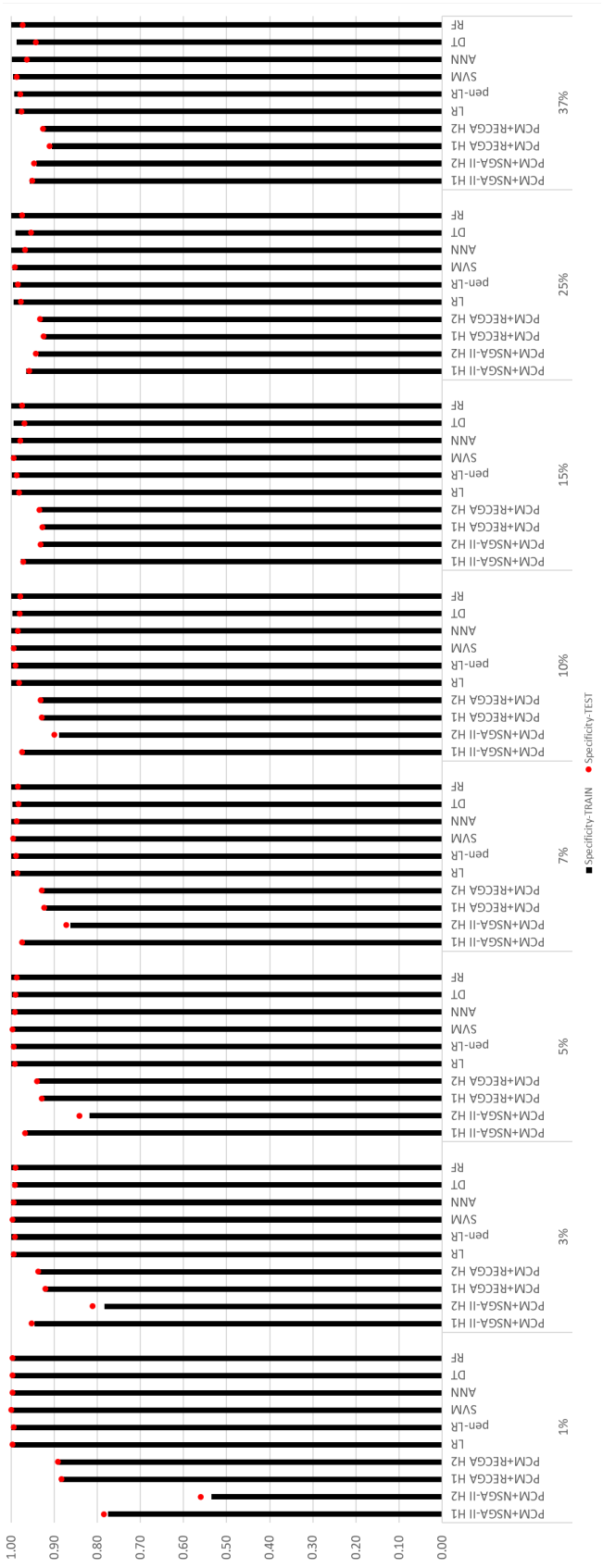


Figure 5.4: Training vs. Test Performances (WBCD Dataset)

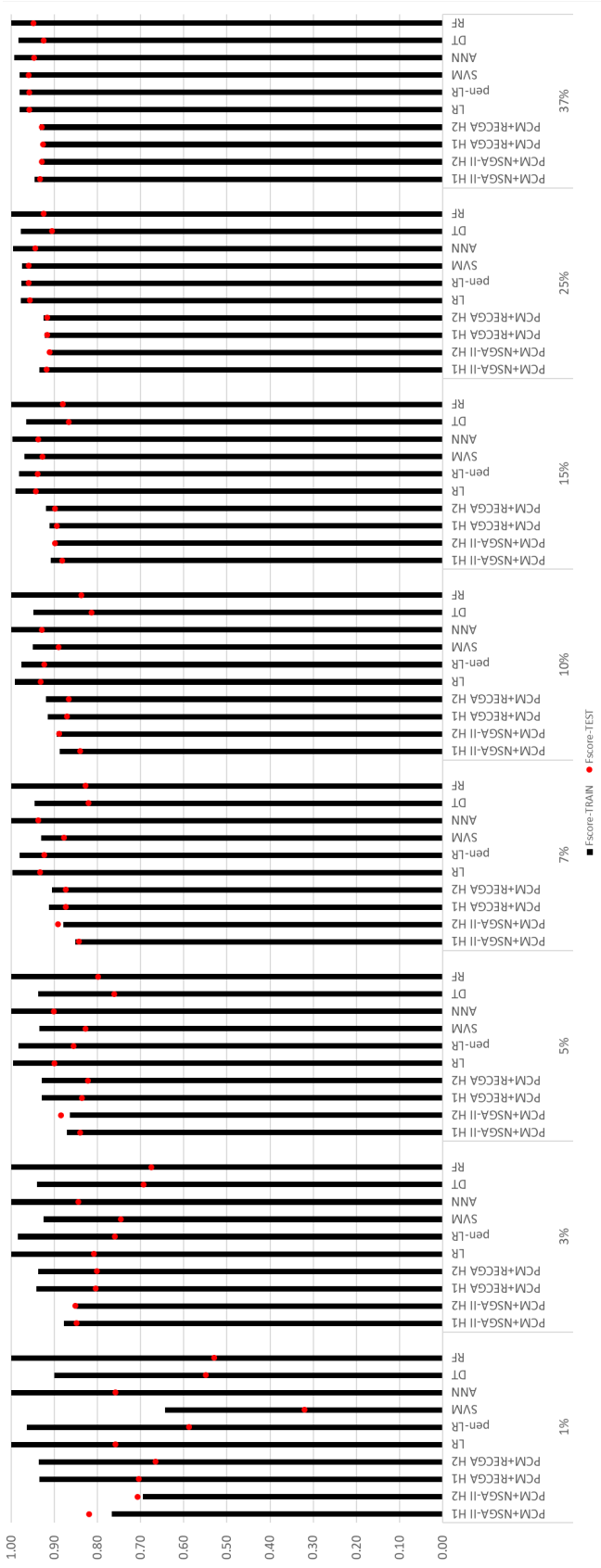


Figure 5.4: Training vs. Test Performances (WBCD Dataset)

It is observed that, for all rarity levels, the specificity of competitor models, both in training and test, are extremely high. However, for the configurations with low rareness levels, although some competitor models have very high sensitivity performances in training, their values in test are poor. For example, for rareness level of 1%, LR, ANN and RF have sensitivity of 1.00 in training but their performances in test are 0.76, 0.76 and 0.53, respectively. Fscore performances of the competitor models reflect the incompatibility between training and test performances, as well. Especially for low levels of rarity, although the training performances of the competitor models are high, their Fscore in test are not as promising as their training results. For example, while the training Fscore values of LR, pen-LR, ANN, DT and RF are 1.00, 0.96, 1.00, 0.90 and 1.00 their test results are 0.76, 0.59, 0.76, 0.55 and 0.53, respectively.

Figure 5.5 shows the gap between training and test performances and average gap of the models. It is observed that, for all models, the gap between training and test is quite small for specificity. Figure 5.5 also suggests that, as the rareness level grows, the corresponding gaps diminish for sensitivity and Fscore. Due to the fact that, since more observations are utilized in training, models learn the specifications of both classes better and their generalization error (test error) decreases.

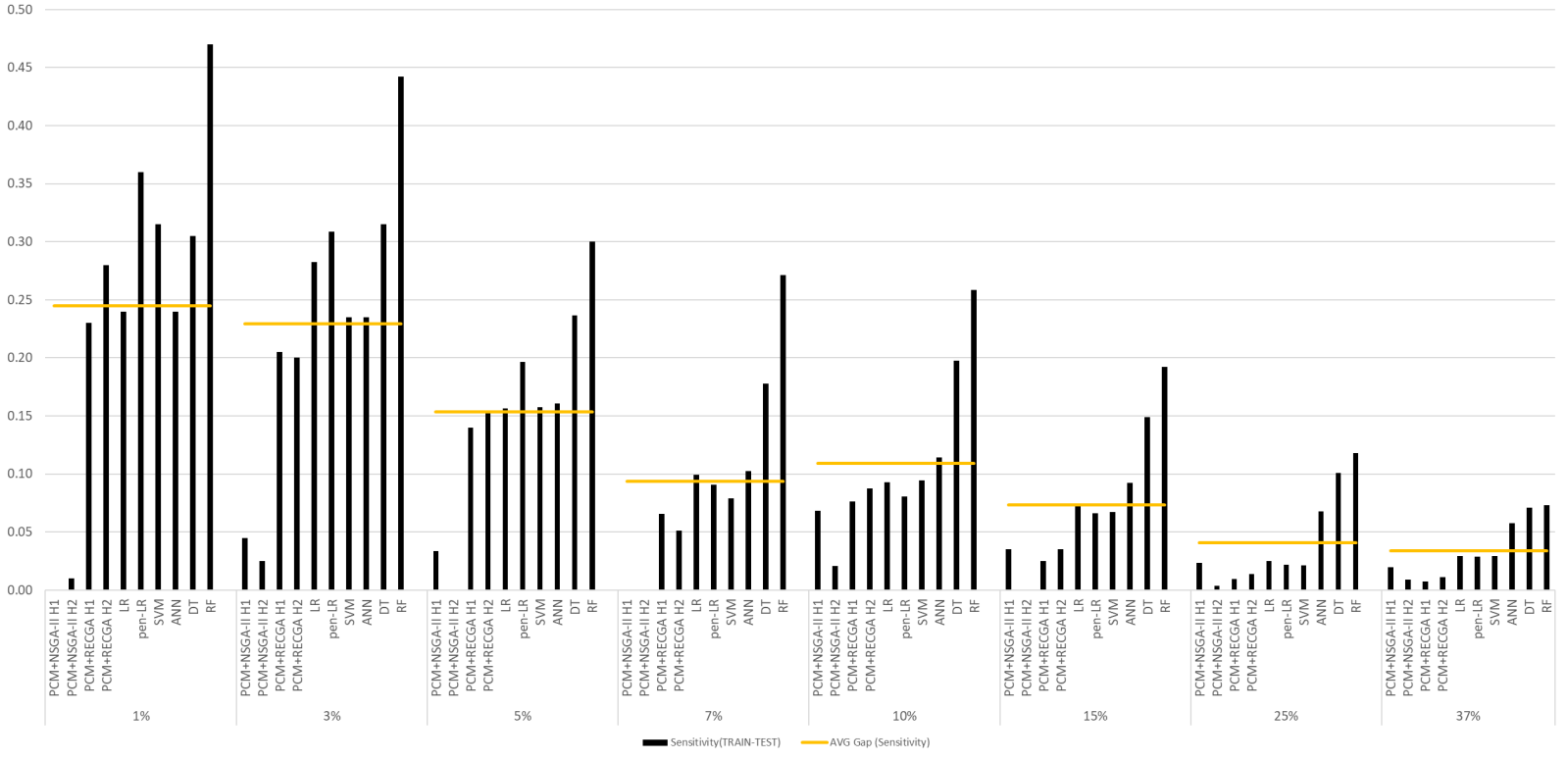


Figure 5.5: Gap Between Training and Test Performances (WBCD Dataset)

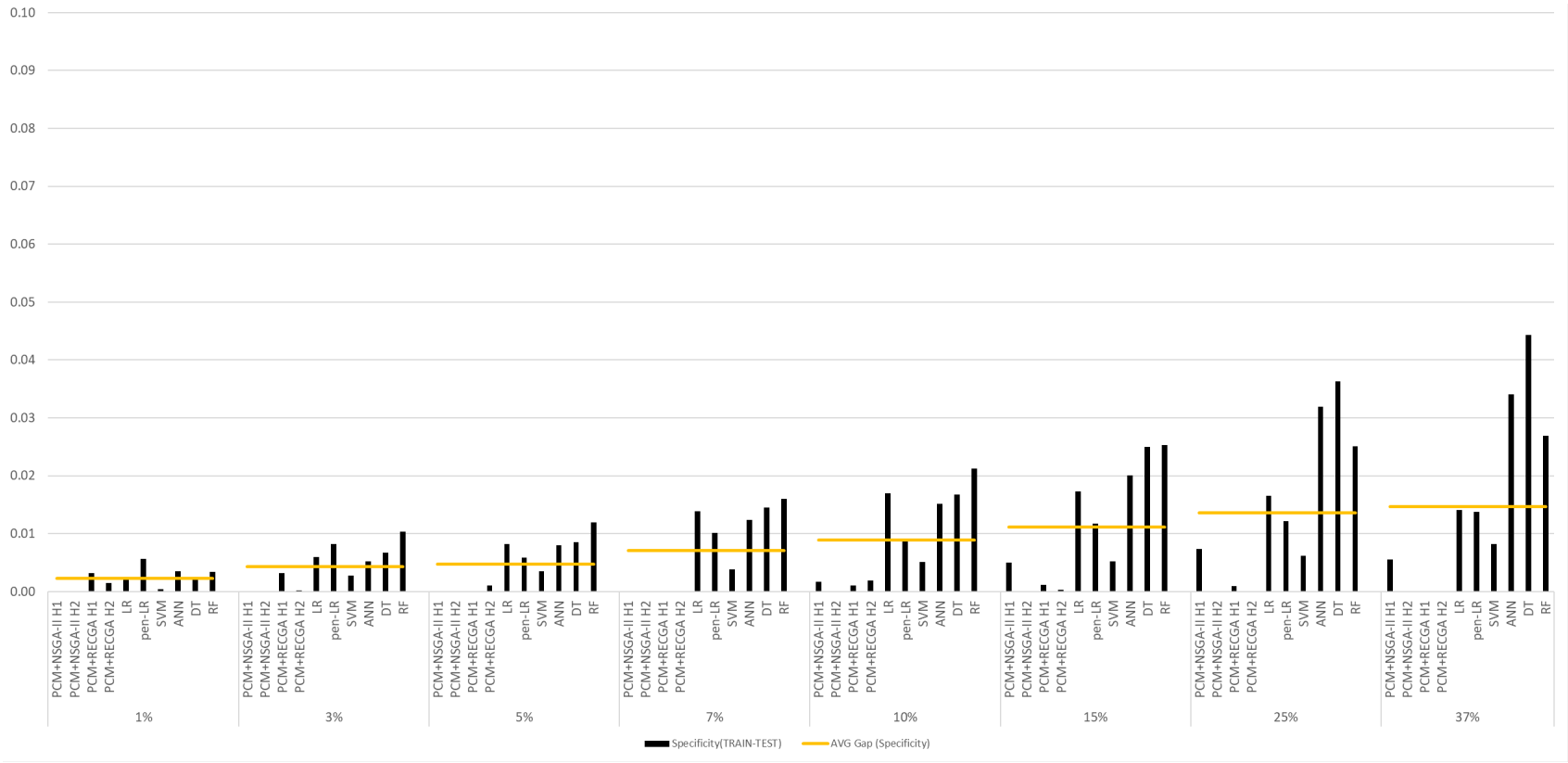


Figure 5.5: Gap Between Training and Test Performances (WBCD Dataset)

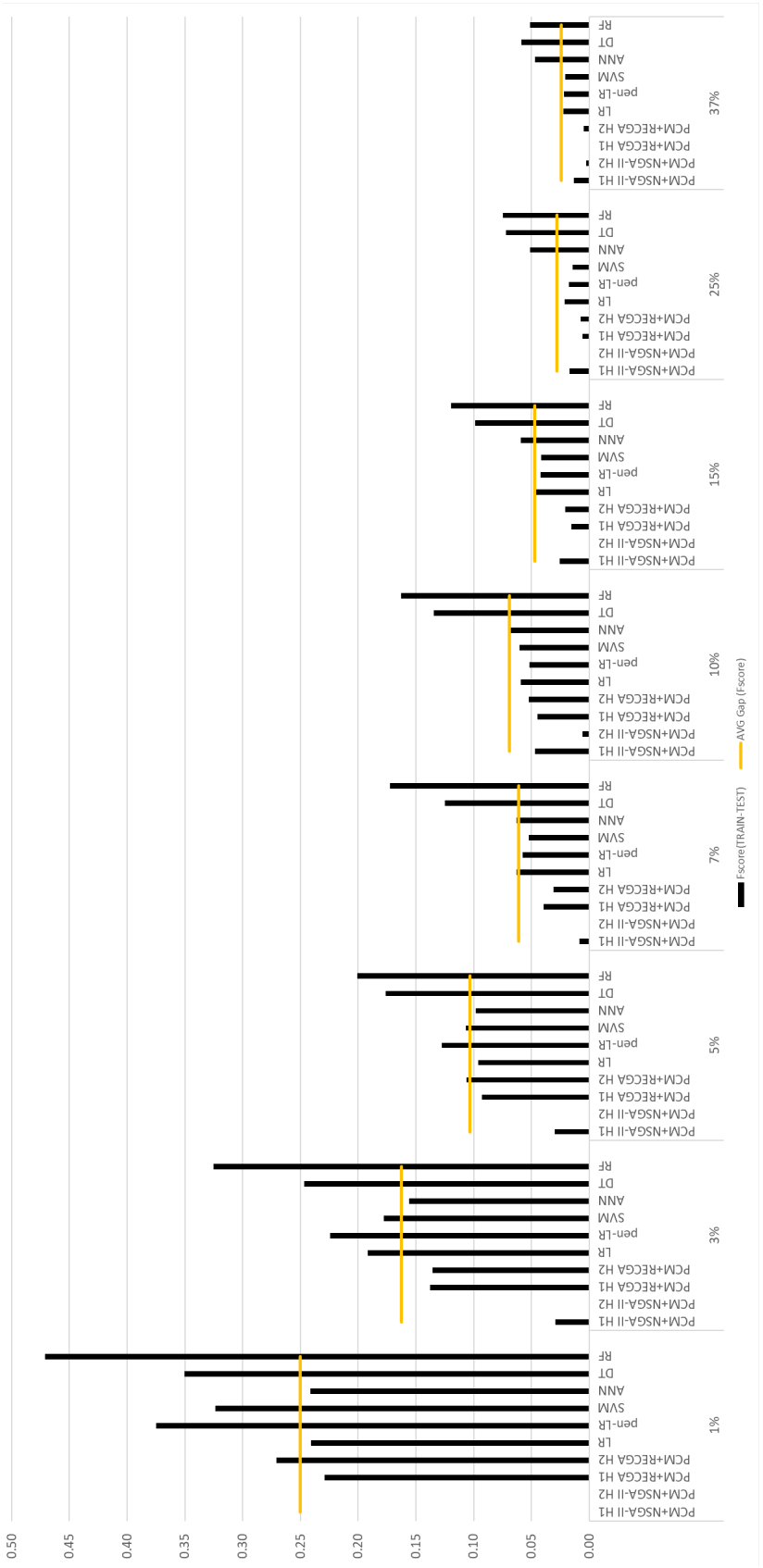


Figure 5.5: Gap Between Training and Test Performances (WBCD Dataset)

For the rareness levels less than 10% (i.e. 1%, 3%, 5%, 7%), the only models whose gap between training and test performances are always below the average, in terms of sensitivity and Fscore, are PCM+NSGA-II (regardless of whether the hyperparameter set is H1 or H2) and PCM+RECGA^{H1}. For the remaining configurations (i.e. 10%, 15%, 25%, 37%), the gap between training and test performances of LR, pen-LR and SVM, along with the PCM+NSGA-II and PCM+RECGA, are below the average gap, as well. In addition to the fact that the test performances of PCM+NSGA-II and PCM+RECGA are compatible with those of training, they provide high classification performances. For 1%, 3%, 5% and 7% rareness levels, Fscore of PCM+NSGA-II^{H1} and PCM+RECGA^{H1} are 0.82, 0.85, 0.84 and 0.84 and 0.70, 0.80, 0.84 and 0.87, respectively. For the remaining configurations (10%, 15%, 25%, 37%), Fscore values of PCM+NSGA-II^{H2} and PCM+RECGA^{H2} are 0.89, 0.90, 0.91 and 0.93 and 0.87, 0.90, 0.92 and 0.93, respectively.

Therefore, it can be claimed that, PCM+NSGA-II and PCM+RECGA are strong alternatives with high generalization ability, especially when one class of observations are rare compared to other.

The graphics in Figure 5.6 give the average test performances and standard deviations of sensitivity, specificity and Fscore. As it can be observed from Table 5.30 and Figure 5.6, the performance of PCM+RECGA does not change significantly whether the experiments are conducted with hyper-parameter sets H1 or H2. The same analysis for PCM+NSGA-II suggests that, in general, the model performances are similar for H1 and H2. However, when the positive cases are extremely rare, selecting optimal model parameters provides great advantage. That is, in 1% of rareness level, PCM+NSGA-II^{H1} has far better Fscore value than that of PCM+NSGA-II^{H2}. Note that, we expect that, the model performances could improve if the hyper-parameter optimization is conducted for each configuration of rareness.

The graphics also indicate that, when the rareness level is higher than 10%, the Fscore of PCM+NSGA-II^{H2} and PCM+RECGA^{H2} are not worse than 0.90. The competitor models have Fscore values between 0.87 and 0.96 in these configurations. Thus, we can claim that, the developed models are able to compete with well-known machine learning models, when the distribution of positive and negative observations are bal-

anced and when one class of observations becomes extremely rare in the population, the suggested models outperforms most of the competitor models.

In 1% of rarity, the best Fscore values belong to PCM+NSGA-II^{H1} (0.82), LR (0.76), ANN (0.76) and PCM+RECGA^{H1} (0.70), respectively. Their closest rival is pen-LR which is far behind them with an Fscore value of 0.59. In 3% of rareness, the ranking remains the same. PCM+NSGA^{H1} is the best model with its Fscore value of 0.85 and it is followed by ANN (0.84), LR (0.81) and PCM+RECGA^{H1} (0.80), respectively. In rareness level of 5%, the competitor models' Fscore ranges between 0.76 and 0.90, where the performances of the proposed models (PCM+NSGA-II^{H1} and PCM+NSGA-II^{H2}) are 0.84. When the rarity becomes 7%, Fscore performances of the competitor models are between 0.82 and 0.93. PCM+RECGA^{H1} has an Fscore of 0.87 and the same performance indicator of PCM+NSGA-II^{H1} is 0.84. Finally, in 10% of rareness, Fscore values of PCM+NSGA-II^{H2} and PCM+RECGA^{H2} are 0.89 and 0.87, respectively, where the competitor models' values are between 0.81 and 0.93.

These results indicate that, PCM+NSGA-II and PCM+RECGA are promising classification algorithms, and they are more robust, compared to some well-known machine learning algorithms, especially under the conditions of class imbalance.

The graphics in Figure 5.6 also indicate that, for high levels of rareness, the standard deviations are small and as the proportion of positive cases decreases in the population, standard deviations of sensitivity and Fscore of all models tend to grow. PCM+NSGA-II is one of the most robust models where the standard deviations do not change significantly for the cases with low rarity levels. On the other hand, especially for the competitor models, a significant amount of increase is observed in standard deviation.

As in the previous cases, we do not give a detailed analysis about Fmeasure performances since it is not one of our performance indicators and it does not have an interpretation for the configurations where one class of observations are rare compared to other. However, it can be said that, in terms of Fmeasure, the proposed models are able to compete with the competitor models for the configurations where the positive and negative observations have a relatively balanced distribution. Since

all the competitor models solve an instance within a minute, we do not report their solution times in detail.

The comparison of the proposed models, PCM+NSGA-II and PCM+RECGA indicates that, except the extremely rare cases (1% and 3%), models have close performances. However, in these configurations, PCM+NSGA-II^{H1} has higher average performances and lower standard deviations in Fscore than those of PCM+RECGA^{H1}.

For detailed tables that give performances of the suggested and the compared models see the Appendix (Section L). The tables also give the number of correct classifications as well as ratio of correct classifications.

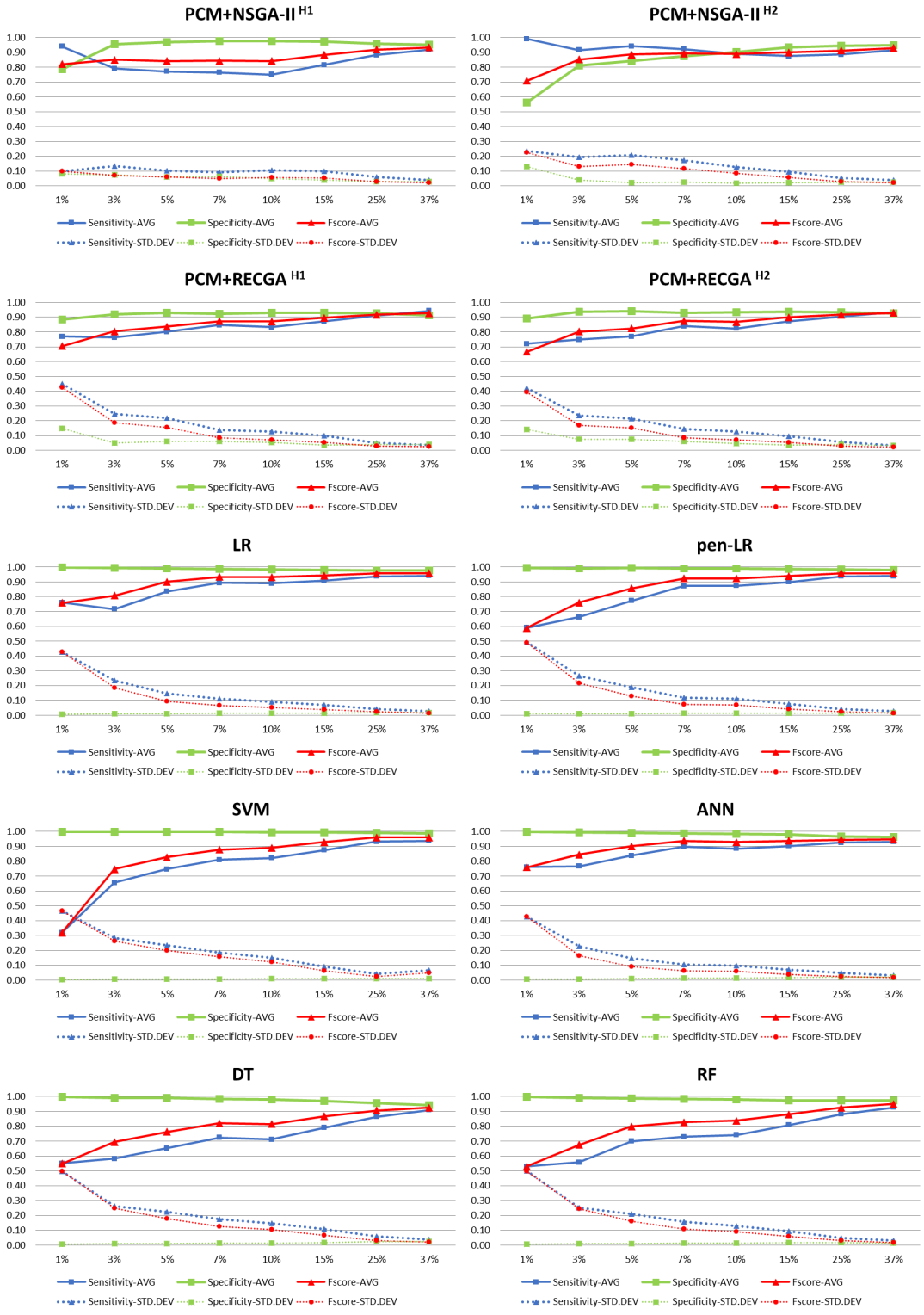


Figure 5.6: Performances for Different Rareness Levels (WBCD Dataset)

When the model performances are compared with those of the models that exist in the literature, among studies discussed in Section 5.2, only two of them ([1], [2]) consider class imbalance with 9% rarity. In the original data, the accuracy results reported in the literature varies between 92.97% and 99.04%, and for most of them, sensitivity and specificity rates are above 90%. In the corresponding setting (rareness level = 37%) accuracy, sensitivity and specificity performances of PCM+NSGA-II^{H2} are 93%, 91% and 95%, respectively, while the same performance indicators are 93% for PCM+RECGA^{H2}.

When we analyze the studies that test their models with a rarity around 10%, it is observed that maximum accuracy is 92.55%. The best sensitivity and specificity rates are above 87% and 96%, respectively. Accuracy, sensitivity and specificity values are 90%, 89%, 90% and 92%, 82%, 93% for PCM+NSGA-II^{H2} and PCM+RECGA^{H2}, respectively. Thus, it can be seen that, our models can compete with successful models of the literature.

Finally, when we compare the performances of models in WBCO and WBCD datasets, it is observed that, no specific model shows different performances for these two sets of data. However, in general, while the competitor models perform better in WBCD dataset than WBCO, for PCM+NSGA-II and PCM+RECGA the opposite is true. One of the explicit differences of WBCO and WBCD datasets is that, while the former consists of categorical factors, the latter has continuous factor values. Therefore, the reason behind the observation mentioned above may be better performance of PCM+NSGA-II and PCM+RECGA in categorical datasets.

5.6 Conclusion

In medical diagnosis problems, the goal is to achieve high sensitivity and specificity, simultaneously. As one class of observations becomes rare compared to the other, classification becomes harder. If the negative (positive) observations constitute the majority, standard classification algorithms achieve high specificity (sensitivity) rates but their ability in sensitivity (specificity) stays limited. In this study, by integrating a Mixed-Integer Linear Model, PCM, with evolutionary algorithms, we develop

promising classification models, which are robust in the existence of significant class imbalance.

Using WBCO and WBCD datasets, we conduct numerical experiments with eight levels of rarity of positive observations, ranging between 1% to 35% and 1% to 37%, respectively. The experimental analysis suggests that, generating initial solutions of the evolutionary algorithms with PCM yields better results than random generation of initial solutions. Then, the performances of suggested models are compared with several well-known machine learning methods. It is observed that, proposed models are promising classification algorithms and they are stronger when one class of observations are rare compared to other. In other words, as the malignant tumors become rare in the population, although the competitor models lose their ability of detecting positive (malignant) cases, PCM+NSGA-II and PCM+RECGA preserve their capability of detecting positive observations without sacrificing too much from their high detection ability of negative cases.

However, we must note that, the competitor models applied in this study are in the most classical forms suggested in the literature. That is, they are not designed specifically for rare events. For example, determining the hyper-parameters of the competitor models, such as the kernel parameter in SVM or the number of hidden layers in ANN, with the aim of good performances in rare cases (i.e. high Fscore) may improve the performance of these models when one class of observations are rare compared to other.

The comparison of PCM+NSGA-II and PCM+RECGA suggests that, they have close performances in general. However, in the extremely rare cases, PCM+NSGA-II perform better than PCM+RECGA with its higher average results and lower standard deviations. On the other hand, PCM+RECGA is more robust against the changes in hyper-parameter choices. According to results of our experiments, we can claim that, both models are able to compete with the studies existing in the literature.

CHAPTER 6

CONCLUSION

It is very important to estimate the presence or absence of a disease. There are various medical diagnostic methods used for this purpose. However, these methods are often expensive and/or risky for the patients. Moreover, in some cases no obvious symptoms or clinical signs are observed in the presence of a disease. Instead of using expensive diagnostic methods or clinical tests, decision support tools employing operations research techniques and machine learning methods can be developed. They can certainly help medical doctors make true diagnostic predictions without creating additional risks and costs for the patients.

In this study, we aim to develop methods which classify patients in two categories of disease status correctly. For this purpose, we develop hybrid methods which integrates multi-criteria decision making, evolutionary algorithms and machine learning. The suggested classification algorithms are designed to be used by medical experts as a decision support tool. We give priority to obtaining high sensitivity, provided that the specificity values are also reasonable. Thus, we aim to minimize the risks associated with human life in return for financial burden resulting from further investigation.

Since the proposed methodologies lie in the intersection of many disciplines, we provide a broad literature review, which includes an overview of machine learning, prediction models in health-care, multi-criteria decision analysis, rare event classification and role of evolutionary algorithms in machine learning and multi-objective decision analysis.

First, we develop a Mixed-Integer Linear Programming model, PCM. It is a variant

of multi-criteria decision analysis method, UTADIS, and we specifically design it for medical classification problems. After solving PCM with various values of a specific parameter, we obtain a set of solutions spread over the Pareto-optimal front in the two dimensional space of true positive and true negative responses. That is, it creates a set of solutions where some of them have the capacity to achieve high sensitivity while some have the capacity to achieve high specificity. Then, we integrate PCM with evolutionary algorithms, such that, by tuning the solutions (parameters) obtained from PCM, they aim to find hybrid solutions that can have high sensitivity and specificity, simultaneously. These methods provide more precise classification of the patients in accordance with the specified classification objectives.

The first evolutionary algorithm integrated with PCM is NSGA-II. It is a multi-objective evolutionary algorithm that prefers non-dominated solutions to be transferred to the next generation. This method aims to obtain solutions whose true positive and true negative classification performances are good, simultaneously. The proposed algorithm is named as PCM+NSGA-II. Then, we develop another classification algorithm, RECGA, to integrate with PCM. It only favors solutions whose sensitivity and specificity are simultaneously high. Therefore, the fitness function of the algorithm evaluates a solution by its Fscore value. The suggested algorithm is called as PCM+RECGA. The experimental analyses of these methods are performed on three medical datasets.

Before starting the experimental analyses, we perform hyper-parameter tuning to choose the set of optimal hyper-parameter values for each model and we repeat this process for each datasets.

The problem with the first dataset addresses patient classification considering the risk of restenosis after coronary stent implantation. The objective of this study is to classify patients according to the in-stent-restenosis risk utilizing patient, disease, procedure and lesion related parameters to support doctors in their diagnosis decision. In this context, we first determine the related predictors by investigating the relevant medical literature and consulting with experts. Then, we apply feature selection to find the most related predictors to build the most effective model in prediction ability. The response is the cardiac restenosis status of the patients which indicates whether a

restenosis is expected to exist or not within the period of 36-month beginning with a coronary stent implantation. We gather the data by scanning the records of 10,435 patients between the years 2005 and 2016. 303 observations are found eligible. We test the models' performances through two different settings, where the ratio of positive and negative observations in training, validation and test samples change.

In order to observe the effectiveness of integrating evolutionary algorithms with PCM, we compare the performances of PCM+NSGA-II and PCM+RECGA with the models where the initial solution sets are randomly generated. Then, we compare the proposed models' performances with widely known machine learning methods (competitor models). It is observed that, the models whose initial solution set are randomly generated have biased and highly unbalanced classification results. Thus, they are not reliable and it is clearly better to generate the initial solutions of the evolutionary algorithms with the Mixed-Integer Linear model, PCM. It is also observed that, classification performances of the models are affected by the amount of positive and negative observations used for training. We have seen that, the proposed models are more robust than the competitor models. Additionally, if there are relatively few amount of positive observations and it is more important to correctly identify the presence of a disease, keeping the number of positive and negative observations equal in training sets yields higher sensitivity rates. Furthermore, the experimental analysis suggests that, PCM+NSGA-II and PCM+RECGA have high training and test performances, which indicates their generalization ability. It is also worth to note that, sensitivity, specificity, and accuracy rates of PCM+NSGA-II and PCM+RECGA are promising, compared to the clinical detection methods used by medical experts. Additionally, the proposed algorithms provide great advantage to the medical experts to foresee the risk of in-stent-restenosis at the time of the stent implantation operation.

Then, to show the efficiency of the models from another aspect, we have designed an experimental setting to compare classification performances of the proposed models with a group of medical doctors who are specialized in the area of coronary in-stent-restenosis. 15 cardiologists have participated in this study.

The results of the analyses suggest that, PCM+NSGA-II and PCM+RECGA are reliable and effective decision support techniques for cardiologists in determining poten-

tial restenosis status of a patient.

In the second part of our study, we test the proposed models in two, well studied, structured, large size datasets: WBCO and WBCD datasets. To see the classification performances of the proposed models when the incidence of the disease among the population is low, we perform experimental analysis by creating class imbalance between malignant and benign tumors. The datasets are preprocessed by eliminating correlated factors. We also conduct feature elimination on WBCD dataset.

We repeat the analysis about effectiveness of generating initial solutions of evolutionary algorithms with PCM and we find that, PCM yields better results than randomly generating. Then, we compare the performances of PCM+NSGA-II and PCM+RECGA with those of competitor models.

We conclude that, PCM+NSGA-II and PCM+RECGA are good classification algorithms that can compete with well-known machine learning models when observations of both classes under consideration are close to each other. On the other hand, when one class of observations becomes extremely rare, the proposed models outperform most of the competitor models. It is also observed that, although PCM+RECGA is more robust to the changes in hyper-parameters, conducting the experiments of PCM+NSGA-II with the hyper-parameter values specifically determined to the given rarity level gives better results (especially in low levels of rarities), in general. The experimental analysis suggests that, PCM+NSGA-II and PCM+RECGA mostly have close performances, but in the cases where rarity level is extremely low, PCM+NSGA-II performs better. In comparison of competitor models and proposed algorithms, it is observed that, while the former perform better in WBCD dataset, the latter has higher performances in WBCO dataset. Thus, we conclude that, the proposed models may be better options when the dataset comprised of categorical factors.

In summary, we develop two hybrid methods which integrates multi-criteria decision making, evolutionary algorithms and machine learning. Both models are applied on real data and the experimental analyses suggest that they are promising classification models that can be used by medical experts as decision support tools. One of the distinguishing features of these models is their flexibility. Suggested models focus

on problems where response is represented with a dichotomous variable, however, they can be extended to perform classification in the existence of multiple classes, as well. The models give the classification decision through majority voting, which makes them robust to the variations either in data or generated solution set. Variations of the models can be developed by changing this to consensus voting or assigning a threshold to the number of votes to win. Additionally, in predicting the class of a specific individual, a decision mechanism which assigns probabilities of risk rather than giving binary decisions can be developed. In this case, the risk of the existence of the disease can be determined based on the amount of difference between global utilities that correspond to positive and negative classifications. Integration of feature selection process and the models can be another extension of this study. In this way, it may be possible to make classification decisions faster and more accurate. Finally, integration of PCM with some other evolutionary algorithms can also be considered as a future research direction.

REFERENCES

[1] Y. M. Huang and S. X. Du, "Weighted support vector machine for classification with uneven training class sizes," in *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, vol. 7, pp. 4365–4369, IEEE, 2005.

[2] S. X. Du and S. T. Chen, "Weighted support vector machine for classification," in *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, vol. 4, pp. 3866–3871, IEEE, 2005.

[3] M. Doumpos and C. Zopounidis, "On the use of a multi-criteria hierarchical discrimination approach for country risk assessment," *Journal of Multi-Criteria Decision Analysis*, vol. 11, no. 4-5, pp. 279–289, 2002.

[4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[5] W. H. Wolberg, O. L. Mangasarian, and D. W. Aha, "UCI machine learning repository, breast cancer wisconsin (original) data set." `https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Original)`, 1992.

[6] W. H. Wolberg, N. W. Street, and O. L. Mangasarian, "UCI machine learning repository, breast cancer wisconsin (diagnostic) data set." `https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)`, 1995.

[7] H. Şahin, S. Duran, E. Yakıcı, and M. Şahin, "Patient classification considering the risk of restenosis after coronary stent placement," *Journal of Heuristics*, vol. 25, no. 4-5, pp. 703–729, 2019.

[8] E. Alpaydin, *Introduction to machine learning*. MIT press, 2014.

[9] J. G. Carbonell, R. S. Michalski, and T. M. Mitchell, "An overview of machine learning," in *Machine learning*, pp. 3–23, Springer, 1983.

[10] T. M. Mitchell, "Machine learning. 1997," *Burr Ridge, IL: McGraw Hill*, vol. 45, no. 37, pp. 870–877, 1997.

[11] B. D. Ripley, *Pattern recognition and neural networks*. Cambridge university press, 2007.

[12] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, pp. 3–24, 2007.

[13] R. H. Lin, "An intelligent model for liver disease diagnosis," *Artificial Intelligence in Medicine*, vol. 47, no. 1, pp. 53–62, 2009.

[14] F. M. E. Uzoka, J. Osuji, and O. Obot, "Clinical decision support system (dss) in the diagnosis of malaria: A case comparison of two soft computing methodologies," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1537–1553, 2011.

[15] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[16] M. J. Huang, M. Y. Chen, and S. C. Lee, "Integrating data mining with case-based reasoning for chronic diseases prognosis and diagnosis," *Expert systems with applications*, vol. 32, no. 3, pp. 856–867, 2007.

[17] J. Han, M. Kamber, and A. K. Tung, "Spatial clustering methods in data mining," *Geographic data mining and knowledge discovery*, pp. 188–217, 2001.

[18] N. Freed and F. Glover, "Simple but powerful goal programming models for discriminant problems," *European Journal of Operational Research*, vol. 7, no. 1, pp. 44–60, 1981.

[19] Z. Zhang, Y. Shi, and G. Gao, "A rough set-based multiple criteria linear programming approach for the medical diagnosis and prognosis," *Expert Systems with Applications*, vol. 36, no. 5, pp. 8932–8937, 2009.

[20] T. M. Mitchell, "Machine learning and data mining," *Communications of the ACM*, vol. 42, no. 11, 1999.

[21] B. Venkatalakshmi and M. Shivsankar, "Heart disease diagnosis using predictive data mining," *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 3, no. 3, pp. 1873–7, 2014.

[22] P. Mangiameli, D. West, and R. Rampal, "Model selection for medical diagnosis decision support systems," *Decision Support Systems*, vol. 36, no. 3, pp. 247–259, 2004.

[23] J. Nahar, T. Imam, K. S. Tickle, and Y. P. P. Chen, "Computational intelligence for heart disease diagnosis: A medical knowledge driven approach," *Expert Systems with Applications*, vol. 40, no. 1, pp. 96–104, 2013.

[24] M. Jabbar, B. Deekshatulu, and P. Chandra, "Computational intelligence technique for early diagnosis of heart disease," in *2015 IEEE International Conference on Engineering and Technology (ICETECH)*, pp. 1–6, IEEE, 2015.

[25] M. Neshat, M. Sargolzaei, A. Nadjaran Toosi, and A. Masoumi, "Hepatitis disease diagnosis using hybrid case based reasoning and particle swarm optimization," *ISRN Artificial Intelligence*, vol. 2012, 2012.

[26] W. Du and Z. Zhan, "Building decision tree classifier on private data," in *Proceedings of the IEEE international conference on Privacy, security and data mining-Volume 14*, pp. 1–8, Australian Computer Society, Inc., 2002.

[27] R. Rastogi and K. Shim, "Public: A decision tree classifier that integrates building and pruning," in *VLDB*, vol. 98, pp. 24–27, 1998.

[28] W. J. Kuo, R. F. Chang, D. R. Chen, and C. C. Lee, "Data mining with decision trees for diagnosis of breast tumor in medical ultrasonic images," *Breast cancer research and treatment*, vol. 66, no. 1, pp. 51–57, 2001.

[29] S. Palaniappan and R. Awang, "Intelligent heart disease prediction system using data mining techniques," in *2008 IEEE/ACS international conference on computer systems and applications*, pp. 108–115, IEEE, 2008.

[30] A. A. Al Jarullah, "Decision tree discovery for the diagnosis of type ii diabetes," in *2011 International conference on innovations in information technology*, pp. 303–307, IEEE, 2011.

[31] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[32] S. L. A. Lee, A. Z. Kouzani, and E. J. Hu, "Random forest based lung nodule classification aided by clustering," *Computerized medical imaging and graphics*, vol. 34, no. 7, pp. 535–542, 2010.

[33] D. West, P. Mangiameli, R. Rampal, and V. West, "Ensemble strategies for a medical diagnostic decision support system: A breast cancer diagnosis application," *European Journal of Operational Research*, vol. 162, no. 2, pp. 532–551, 2005.

[34] B. Malmir, M. Amini, and S. I. Chang, "A medical decision support system for disease diagnosis under uncertainty," *Expert Systems with Applications*, vol. 88, pp. 95–108, 2017.

[35] G. Zhang, B. E. Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks:: The state of the art," *International journal of forecasting*, vol. 14, no. 1, pp. 35–62, 1998.

[36] Y. Wu, M. L. Giger, K. Doi, C. J. Vyborny, R. A. Schmidt, and C. E. Metz, "Artificial neural networks in mammography: application to decision making in the diagnosis of breast cancer.," *Radiology*, vol. 187, no. 1, pp. 81–87, 1993.

[37] G. Carrault, M.-O. Cordier, R. Quiniou, and F. Wang, "Temporal abstraction and inductive logic programming for arrhythmia recognition from electrocardiograms," *Artificial intelligence in medicine*, vol. 28, no. 3, pp. 231–263, 2003.

[38] D. Conforti and R. Guido, "Kernel based support vector machine via semidefinite programming: Application to medical diagnosis," *Computers & Operations Research*, vol. 37, no. 8, pp. 1389–1394, 2010.

[39] D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, and M. Klein, *Logistic regression*. Springer, 2002.

[40] M. Y. Park and T. Hastie, "Penalized logistic regression for detecting gene interactions," *Biostatistics*, vol. 9, no. 1, pp. 30–50, 2007.

[41] D. Firth, "Bias reduction of maximum likelihood estimates," *Biometrika*, vol. 80, no. 1, pp. 27–38, 1993.

[42] G. King and L. Zeng, "Logistic regression in rare events data," *Political analysis*, pp. 137–163, 2001.

[43] O. L. Mangasarian, W. N. Street, and W. H. Wolberg, "Breast cancer diagnosis and prognosis via linear programming," *Operations Research*, vol. 43, no. 4, pp. 570–577, 1995.

[44] T. L. Saaty, "Analytic hierarchy process," *Encyclopedia of Biostatistics*, vol. 1, 2005.

[45] M. J. Liberatore and R. L. Nydick, "The analytic hierarchy process in medical and health care decision making: A literature review," *European Journal of Operational Research*, vol. 189, no. 1, pp. 194–207, 2008.

[46] P. R. Pinheiro, A. K. A. de Castro, and M. C. D. Pinheiro, "A multicriteria model applied in the diagnosis of alzheimer's disease: a bayesian network," in *2008 11th IEEE International Conference on Computational Science and Engineering*, pp. 15–22, IEEE, 2008.

[47] A. T. Brasil Filho, P. R. Pinheiro, A. L. Coelho, and N. C. Costa, "Comparison of two mcda classification methods over the diagnosis of alzheimer's disease," in *International Conference on Rough Sets and Knowledge Technology*, pp. 334–341, Springer, 2009.

[48] C. Zopounidis and M. Doumpos, "Multicriteria classification and sorting methods: a literature review," *European Journal of Operational Research*, vol. 138, no. 2, pp. 229–246, 2002.

[49] S. Greco, J. Figueira, and M. Ehrgott, "Multiple criteria decision analysis," *Springer's International series*, 2005.

[50] C. Zopounidis and M. Doumpos, "Multi-criteria decision aid in financial decision making: methodologies and literature review," *Journal of Multi-Criteria Decision Analysis*, vol. 11, no. 4-5, pp. 167–186, 2002.

[51] M. Doumpos and C. Zopounidis, "Assessing financial risks using a multicriteria sorting procedure: the case of country risk assessment," *Omega*, vol. 29, no. 1, pp. 97–109, 2001.

[52] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[53] C. L. Blake and C. Merz, "UCI machine learning repository." `http://archive.ics.uci.edu/ml`, 1998.

[54] S. Liu, C. Y. Jia, and H. Ma, "A new weighted support vector machine with ga-based parameter selection," in *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, vol. 7, pp. 4351–4355, IEEE, 2005.

[55] "Ida benchmark repository used in several boosting, kfd and svm papers." `http://ida.first.gmd.de/ratsch/data/benchmarks.htm`.

[56] X. Yang, Q. Song, and Y. Wang, "A weighted support vector machine for data classification," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 21, no. 05, pp. 961–976, 2007.

[57] A. Frank and A. Asuncion, "UCI machine learning repository." `http://archive.ics.uci.edu/ml`, 2010.

[58] J. Li, L. S. Liu, S. Fong, R. K. Wong, S. Mohammed, J. Fiaidhi, Y. Sung, and K. K. Wong, "Adaptive swarm balancing algorithms for rare-event prediction in imbalanced healthcare data," *PloS one*, vol. 12, no. 7, p. e0180830, 2017.

[59] N/A, "UCI Machine Learning Repository, KDD Cup 1999 Data Data Set." `https://archive.ics.uci.edu/ml/datasets/kdd+cup+1999+data`, 1999.

[60] M. Bohanec, "UCI Machine Learning Repository, Car Evaluation Data

Set." https://archive.ics.uci.edu/ml/datasets/car+evaluation, 1997.

[61] H. A. Guvenir, B. Acar, and H. Muderrisoglu, "UCI Machine Learning Repository, Arrhythmia Data Set." url-https://archive.ics.uci.edu/ml/datasets/arrhythmia, 1998.

[62] K. Nakai, "UCI Machine Learning Repository, Yeast Data Set." https://archive.ics.uci.edu/ml/datasets/Yeast, 1996.

[63] R. Kohavi and B. Becker, "UCI Machine Learning Repository, AdultData Set." https://archive.ics.uci.edu/ml/datasets/adult, 1996.

[64] J. Catlett, "UCI Machine Learning Repository, Statlog (Shuttle) Data Set." https://archive.ics.uci.edu/ml/datasets/Statlog+(Shuttle), N/A.

[65] W. J. Nash, T. L. Sellers, S. R. Talbot, A. J. Cawthorn, and W. B. Ford, "UCI Machine Learning Repository, Abalone Data Set." https://archive.ics.uci.edu/ml/datasets/abalone, 1995.

[66] K. K. Wankhade, K. C. Jondhale, and V. R. Thool, "A hybrid approach for classification of rare class data," *Knowledge and Information Systems*, vol. 56, no. 1, pp. 197–221, 2018.

[67] B. Deekshatulu, P. Chandra, *et al.*, "Classification of heart disease using k-nearest neighbor and genetic algorithm," *Procedia Technology*, vol. 10, pp. 85–94, 2013.

[68] Y. Goletsis, C. Papaloukas, D. I. Fotiadis, A. Likas, and L. K. Michalis, "Automated ischemic beat classification using genetic algorithms and multicriteria decision analysis," *IEEE transactions on Biomedical Engineering*, vol. 51, no. 10, pp. 1717–1725, 2004.

[69] J. Zhang, Z. H. Zhan, Y. Lin, N. Chen, Y. J. Gong, J. H. Zhong, H. S. Chung, Y. Li, and Y. h. Shi, "Evolutionary computation meets machine learning: A survey," *IEEE Computational Intelligence Magazine*, vol. 6, no. 4, pp. 68–75, 2011.

[70] H. A. Guvenir and E. Erel, "Multicriteria inventory classification using a genetic algorithm," *European journal of operational research*, vol. 105, no. 1, pp. 29–37, 1998.

[71] K. De Jong, "Learning with genetic algorithms: An overview," *Machine learning*, vol. 3, no. 2-3, pp. 121–138, 1988.

[72] G. Kim, D. Seo, and K. I. Kang, "Hybrid models of neural networks and genetic algorithms for predicting preliminary cost estimates," *Journal of Computing in Civil Engineering*, vol. 19, no. 2, pp. 208–211, 2005.

[73] C. L. Huang and C. J. Wang, "A ga-based feature selection and parameters optimizationfor support vector machines," *Expert Systems with applications*, vol. 31, no. 2, pp. 231–240, 2006.

[74] K. J. Kim and I. Han, "Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index," *Expert systems with Applications*, vol. 19, no. 2, pp. 125–132, 2000.

[75] H. Chen, "Machine learning for information retrieval: neural networks, symbolic learning, and genetic algorithms," *Journal of the American society for Information Science*, vol. 46, no. 3, pp. 194–216, 1995.

[76] S. L. Smith, P. Gaughan, D. M. Halliday, Q. Ju, N. M. Aly, and J. R. Playfer, "Diagnosis of parkinson's disease using evolutionary algorithms," *Genetic Programming and Evolvable Machines*, vol. 8, no. 4, pp. 433–447, 2007.

[77] V. Podgorelec and P. Kokol, "Towards more optimal medical diagnosing with evolutionary algorithms," *Journal of Medical Systems*, vol. 25, no. 3, pp. 195–219, 2001.

[78] P. D. Turney, "Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm," *Journal of artificial intelligence research*, vol. 2, pp. 369–409, 1994.

[79] A. L. Corcoran and S. Sen, "Using real-valued genetic algorithms to evolve rule sets for classification," in *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*, pp. 120–124, IEEE, 1994.

[80] S. Das and B. K. Panigrahi, "Multi-objective evolutionary algorithms," in *encyclopedia of artificial intelligence*, pp. 1145–1151, IGI Global, 2009.

[81] K. Deb, *Multi-objective optimization using evolutionary algorithms*, vol. 16. John Wiley & Sons, 2001.

[82] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen, *et al.*, *Evolutionary algorithms for solving multi-objective problems*, vol. 5. Springer, 2007.

[83] E. Zitzler and L. Thiele, "An evolutionary algorithm for multiobjective optimization: The strength pareto approach," 1998.

[84] J. D. Knowles and D. Corne, "Local search, multiobjective optimization and the pareto archived evolution strategy," in *Proceedings of Third Australia-Japan Joint Workshop on Intelligent and Evolutionary Systems*, pp. 209–216, 1999.

[85] D. A. Van Veldhuizen and G. B. Lamont, "Multiobjective optimization with messy genetic algorithms," in *Proceedings of the 2000 ACM symposium on Applied computing-Volume 1*, pp. 470–476, ACM, 2000.

[86] D. G. Conway, A. V. Cabot, and M. Venkataramanan, "A genetic algorithm for discriminant analysis," *Annals of Operations Research*, vol. 78, pp. 71–82, 1998.

[87] W. Zhu, N. Zeng, N. Wang, *et al.*, "Sensitivity, specificity, accuracy, associated confidence interval and roc analysis with practical sas implementations," *NESUG proceedings: health care and life sciences, Baltimore, Maryland*, vol. 19, p. 67, 2010.

[88] A. G. Glaros and R. B. Kline, "Understanding the accuracy of tests with cutting scores: The sensitivity, specificity, and predictive value model," *Journal of clinical psychology*, vol. 44, no. 6, pp. 1013–1023, 1988.

[89] R. Parikh, A. Mathai, S. Parikh, G. C. Sekhar, and R. Thomas, "Understanding and using sensitivity, specificity and predictive values," *Indian journal of ophthalmology*, vol. 56, no. 1, p. 45, 2008.

[90] D. M. Powers, "What the f-measure doesn't measure: Features, flaws, fallacies and fixes," *arXiv preprint arXiv:1503.06410*, 2015.

[91] B. Song, G. Zhang, W. Zhu, and Z. Liang, "Roc operating point selection for classification of imbalanced data with application to computer-aided polyp detection in ct colonography," *International journal of computer assisted radiology and surgery*, vol. 9, no. 1, pp. 79–89, 2014.

[92] B. Song, *ROC Random Forest and Its Application*. PhD thesis, The Graduate School, Stony Brook University: Stony Brook, NY., 2015.

[93] Y. Lan, Q. Wang, J. R. Cole, and G. L. Rosen, "Using the rdp classifier to predict taxonomic novelty and reduce the search space for finding novel organisms," *PLoS one*, vol. 7, no. 3, p. e32491, 2012.

[94] R. Togo, N. Yamamichi, K. Mabe, Y. Takahashi, C. Takeuchi, M. Kato, N. Sakamoto, K. Ishihara, T. Ogawa, and M. Haseyama, "Detection of gastritis by a deep convolutional neural network from double-contrast upper gastrointestinal barium x-ray radiography," *Journal of gastroenterology*, vol. 54, no. 4, pp. 321–329, 2019.

[95] H. Şahin, S. Duran, E. Yakıcı, and M. Şahin, "Computer Code: PCM+NSGA-II." `https://www.researchgate.net/publication/331498700_Computer_Code_PCMNSGA-II`. Accessed: 2019-03-5.

[96] S. Lemm, B. Blankertz, T. Dickhaus, and K.-R. Müller, "Introduction to machine learning for brain imaging," *Neuroimage*, vol. 56, no. 2, pp. 387–399, 2011.

[97] J. Wainer and G. Cawley, "Nested cross-validation when selecting classifiers is overzealous for most practical applications," *arXiv preprint arXiv:1809.09446*, 2018.

[98] B. Bischl, O. Mersmann, and H. Trautmann, "Resampling methods in model validation," in *Workshop on Experimental Methods for the Assessment of Computational Systems (WEMACS 2010), held in conjunction with the Interna-*

tional Conference on Parallel Problem Solving From Nature (PPSN 2010), Krakow, Poland, Sept, vol. 11, p. 14, 2010.

[99] I. Tsamardinos, A. Rakhshani, and V. Lagani, "Performance-estimation properties of cross-validation-based protocols with simultaneous hyper-parameter optimization," *International Journal on Artificial Intelligence Tools*, vol. 24, no. 05, p. 1540023, 2015.

[100] MedStar Health Cardiology Associates, "Angioplasty/stenting." `http://www.heartcapc.com/handler.cfm?event=practice`. Accessed: 2016-02-28.

[101] Patient Education Center, "Coronary artery disease." `http://www.patienteducationcenter.org/articles/coronary-artery-disease/`. Accessed: 2016-02-25.

[102] American Heart Association, "What is a stent?." `https://www.heart.org/idc/groups/heart-public/wcm/hcm/documents/downloadable/ucm300452.pdf`. Accessed: 2016-02-25.

[103] ClevelandClinic, "In-stent restenosis-coronary artery disease." `http://my.clevelandclinic.org/services/heart/disorders/coronary-artery-disease/instentrestenosis`. Accessed: 2016-02-25.

[104] M. J. Price, *Coronary Stenting: A Companion to Topol's Textbook of Interventional Cardiology*. Elsevier Health Sciences, 2013.

[105] A. Doğan, Ö. Kozan, and N. Tüzün, "Stent-içi restenozun fizyopatolojisi ve tedavisi," *Türk Kardiyol Dern Arş*, vol. 33, pp. 115–25, 2005.

[106] G. Dangas and F. Kuepper, "Restenosis: Repeat narrowing of a coronary artery prevention and treatment," *Circulation*, vol. 105, no. 22, pp. 2586–2587, 2002.

[107] R. Van Domburg, D. Foley, P. De Jaegere, P. De Feyter, M. Van den Brand, W. Van der Giessen, J. Hamburger, and P. Serruys, "Long term outcome after coronary stent implantation: a 10 year single centre experience of 1000 patients," *Heart*, vol. 82, no. suppl 2, pp. II27–II34, 1999.

[108] S. Cassese, R. A. Byrne, T. Tada, S. Pinieck, M. Joner, T. Ibrahim, L. A. King, M. Fusaro, K.-L. Laugwitz, and A. Kastrati, "Incidence and predictors of restenosis after coronary stenting in 10 004 patients with surveillance angiography," *Heart*, pp. heartjnl–2013, 2013.

[109] M. Şahin M.D. Cardiologist, "Stents and in-stent-restenosis." Interview on February 28, 2016.

[110] K. Andersen, S. D. Steinþórsdóttir, S. Haraldsdottir, and T. Gudnason, "Clinical evaluation and stress test have limited value in the diagnosis of in-stent restenosis," *Scandinavian Cardiovascular Journal*, vol. 43, no. 6, pp. 402–407, 2009.

[111] G. Dori, Y. Denekamp, S. Fishman, and H. Bitterman, "Exercise stress testing, myocardial perfusion imaging and stress echocardiography for detecting restenosis after successful percutaneous transluminal coronary angioplasty: a review of performance," *Journal of internal medicine*, vol. 253, no. 3, pp. 253–262, 2003.

[112] P. Garzon and M. Eisenberg, "Functional testing for the detection of restenosis after percutaneous transluminal coronary angioplasty: a meta-analysis," *Canadian Journal of Cardiology*, vol. 17, no. 1, pp. 41–48, 2001.

[113] R. B. IstvÃĄnKÃģsa, J. Schneider-Eicke, F. J. Neumann, I. Matsunari, J. Neverve, A. SchÃķmigand, and M. Schwaiger, "Myocardial perfusion scintigraphy to evaluate patients after coronary stent implantation," *J Nucl Med*, vol. 39, pp. 1307–1311, 1998.

[114] Z. Yang, Q. Wang, S. Guo, Y. Zhang, X. Fang, and Z. Cui, "Value of detecting in-stent restenosis by dual source coronary computed tomography coronary angiography," *Zhonghua xin xue guan bing za zhi*, vol. 39, no. 1, pp. 49–52, 2011.

[115] N. Carrabba, J. D. Schuijf, F. R. de Graaf, G. Parodi, E. Maffei, R. Valenti, A. Palumbo, A. C. Weustink, N. R. Mollet, G. Accetta, *et al.*, "Diagnostic accuracy of 64-slice computed tomography coronary angiography for the de-

tection of in-stent restenosis: a meta-analysis," *Journal of Nuclear Cardiology*, vol. 17, no. 3, pp. 470–478, 2010.

[116] T. Gaspar, D. A. Halon, B. S. Lewis, S. Adawi, J. E. Schliamser, R. Rubinshtein, M. Y. Flugelman, and N. Peled, "Diagnosis of coronary in-stent restenosis with multidetector row spiral computed tomography," *Journal of the American College of Cardiology*, vol. 46, no. 8, pp. 1573–1579, 2005.

[117] A. Elhendy, A. F. Schinkel, R. T. van Domberg, J. J. Bax, R. Valkema, and D. Poldermans, "Non-invasive diagnosis of in stent stenosis by stress 99m technetium tetrofosmin myocardial perfusion imaging," *The international journal of cardiovascular imaging*, vol. 22, no. 5, pp. 657–662, 2006.

[118] D. Koller and M. Sahami, "Toward optimal feature selection," tech. rep., Stanford InfoLab, 1996.

[119] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, "Occam's razor," *Information processing letters*, vol. 24, no. 6, pp. 377–380, 1987.

[120] A. G. Lalkhen and A. McCluskey, "Clinical tests: sensitivity and specificity," *Continuing Education in Anaesthesia Critical Care & Pain*, vol. 8, no. 6, pp. 221–223, 2008.

[121] World Cancer Research Fund, "Breast cancer statistics." `https://www.wcrf.org/dietandcancer/cancer-trends/breast-cancer-statistics`. Accessed: 2018-09-17.

[122] Mayo Foundation for Medical Education and Research, "Breast cancer." `https://www.mayoclinic.org/diseases-conditions/breast-cancer/symptoms-causes/syc-20352470`. Accessed: 2019-01-11.

[123] American Cancer Society, "How to detect breast cancer | breast cancer diagnosis." `https://www.cancer.org/cancer/breast-cancer/screening-tests-and-early-detection.html`. Accessed: 2016-02-25.

[124] R. Setiono, "Extracting rules from pruned neural networks for breast cancer diagnosis," *Artificial Intelligence in Medicine*, vol. 8, no. 1, pp. 37–52, 1996.

[125] R. Setiono, "Generating concise and accurate classification rules for breast cancer diagnosis," *Artificial Intelligence in medicine*, vol. 18, no. 3, pp. 205–219, 2000.

[126] C. A. Pena-Reyes and M. Sipper, "A fuzzy-genetic approach to breast cancer diagnosis," *Artificial intelligence in medicine*, vol. 17, no. 2, pp. 131–155, 1999.

[127] J. R. Quinlan, "Improved use of continuous attributes in c4. 5," *Journal of artificial intelligence research*, vol. 4, pp. 77–90, 1996.

[128] H. J. Hamilton, N. Cercone, and N. Shan, *RIAC: a rule induction algorithm based on approximate classification*. Citeseer, 1996.

[129] B. Šter and A. Dobnikar, "Neural networks in medical diagnosis: Comparison with other methods," in *International Conference on Engineering Applications of Neural Networks*, pp. 427–30, 1996.

[130] K. P. Bennett and J. A. Blue, "A support vector machine approach to decision trees," in *Neural Networks Proceedings, 1998. IEEE World Congress on Computational Intelligence. The 1998 IEEE International Joint Conference on*, vol. 3, pp. 2396–2401, IEEE, 1998.

[131] D. Nauck and R. Kruse, "Obtaining interpretable fuzzy classification rules from medical data," *Artificial intelligence in medicine*, vol. 16, no. 2, pp. 149–169, 1999.

[132] D. E. Goodman, L. Boggess, and A. Watkins, "Artificial immune system classification of multiple-class problems," *Proceedings of the artificial neural networks in engineering ANNIE*, vol. 2, pp. 179–183, 2002.

[133] A. A. Albrecht, G. Lappas, S. A. Vinterbo, C. Wong, and L. Ohno-Machado, "Two applications of the lsa machine," in *Neural Information Processing, 2002. ICONIP'02. Proceedings of the 9th International Conference on*, vol. 1, pp. 184–189, IEEE, 2002.

[134] J. Abonyi and F. Szeifert, "Supervised fuzzy clustering for the identification of fuzzy classifiers," *Pattern Recognition Letters*, vol. 24, no. 14, pp. 2195–2207, 2003.

[135] K. Polat and S. Güneş, "Breast cancer diagnosis using least square support vector machine," *Digital signal processing*, vol. 17, no. 4, pp. 694–701, 2007.

[136] E. D. Übeyli, "Implementing automated diagnostic systems for breast cancer detection," *Expert systems with Applications*, vol. 33, no. 4, pp. 1054–1062, 2007.

[137] M. F. Akay, "Support vector machines combined with feature selection for breast cancer diagnosis," *Expert systems with applications*, vol. 36, no. 2, pp. 3240–3247, 2009.

[138] H. L. Chen, B. Yang, J. Liu, and D. Y. Liu, "A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis," *Expert Systems with Applications*, vol. 38, no. 7, pp. 9014–9022, 2011.

[139] D. Lavanya and K. U. Rani, "Ensemble decision tree classifier for breast cancer data," *International Journal of Information Technology Convergence and Services*, vol. 2, no. 1, p. 17, 2012.

[140] K. B. Nahato, K. N. Harichandran, and K. Arputharaj, "Knowledge mining from clinical datasets using rough sets and backpropagation neural network," *Computational and mathematical methods in medicine*, vol. 2015, 2015.

[141] A. M. Abdel Zaher and A. M. Eldeib, "Breast cancer classification using deep belief networks," *Expert Systems with Applications*, vol. 46, pp. 139–144, 2016.

[142] H. Asri, H. Mousannif, H. Al Moatassime, and T. Noel, "Using machine learning algorithms for breast cancer risk prediction and diagnosis," *Procedia Computer Science*, vol. 83, pp. 1064–1069, 2016.

[143] S. Aalaei, H. Shahraki, A. Rowhanimanesh, and S. Eslami, "Feature selection using genetic algorithm for breast cancer diagnosis: experiment on three different datasets," *Iranian journal of basic medical sciences*, vol. 19, no. 5, p. 476, 2016.

[144] A. F. M. Agarap, "On breast cancer detection: an application of machine learning algorithms on the wisconsin diagnostic dataset," in *Proceedings of the 2nd*

International Conference on Machine Learning and Soft Computing, pp. 5–9, ACM, 2018.

[145] W. N. Street, W. H. Wolberg, and O. L. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," in *Biomedical image processing and biomedical visualization*, vol. 1905, pp. 861–871, International Society for Optics and Photonics, 1993.

[146] D. Lavanya and D. K. U. Rani, "Analysis of feature selection with classification: Breast cancer datasets," *Indian Journal of Computer Science and Engineering (IJCSE)*, vol. 2, no. 5, pp. 756–763, 2011.

[147] G. I. Salama, M. Abdelhalim, and M. A.-e. Zeid, "Breast cancer diagnosis on three different datasets using multi-classifiers," *Breast Cancer (WDBC)*, vol. 32, no. 569, p. 2, 2012.

[148] Y. J. Fan and W. A. Chaovalitwongse, "Optimizing feature selection to improve medical diagnosis," *Annals of Operations Research*, vol. 174, no. 1, pp. 169–183, 2010.

[149] Jeff Schneider, "Lecture Notes: Cross Validation." `https://www.cs.cmu.edu/~schneide/tut5/node42.html`. Accessed: 2017-02-25.

APPENDICES

A A Simple Example Explaining PCM+RECGA

Let $|\mathcal{S}| = 2$ where $|\mathcal{S}^-| = 1$ and $|\mathcal{S}^+| = 1$; $|\mathcal{V}| = 2$ where $|\mathcal{V}^-| = 1$ and $|\mathcal{V}^+| = 1$; $|\widetilde{\mathcal{S}}| = 2$ where $|\widetilde{\mathcal{S}}^-| = 1$ and $|\widetilde{\mathcal{S}}^+| = 1$. In other words, training, validation and test samples consist of two observations where one of them is positive and the other one is negative. Assume $|\mathcal{F}| = 2$ and $\mathcal{F} = \{1, 2\}$. i.e. there are two factors.

Recall that O_f is the set of all values of factor f that appear in $\mathcal{S}$. Then, let $|O_1| = 2$ and $|O_2| = 2$. Since both factors that appear in $\mathcal{S}$ take two different values, there is one interval for each factor. Then, the decision variables of are w_{11}, m_{11} and w_{21}, m_{21}.

Since training set has only one negative observation, then false negative classification allowance, L, can be 0 and 1. Thus PCM is solved for $L = 0$ and $L = 1$.

Let (W^0, M^0) be the solution obtained from $PCM(0)$, and (W^1, M^1) be the solution obtained from $PCM(1)$. Then $\mathcal{X} = \{(W^0, M^0), (W^1, M^1)\}$. Also, let *PopulationSize* be 5. Then using (W^0, M^0) and (W^1, M^1) as parents, the algorithm generates a new offspring by performing genetic operations. Let the new solution be (W^2, M^2). Now, $\mathcal{X} = \{(W^0, M^0), (W^1, M^1), (W^2, M^2)\}$. Then, by selecting randomly two parents from $\mathcal{X}$, the algorithm generates another offspring, say (W^3, M^3). Now, $\mathcal{X} = \{(W^0, M^0), (W^1, M^1), (W^2, M^2), (W^3, M^3)\}$. The process is repeated one more time, that is, randomly selected two parents generates an offspring, (W^4, M^4). Now the predetermined *PopulationSize* is reached and $\mathcal{X} = \{(W^0, M^0), (W^1, M^1), (W^2, M^2), (W^3, M^3), (W^4, M^4)\}$.

For each $(W^i, M^i) \in \mathcal{X}$ and for patients in $\mathcal{V}$, $U(p)$ and $\widetilde{U}(p)$ values are calculated to classify patients in the validation set. Then, according to the classification performance of the model in the validation set, sensitivity, specificity and Fscore values

of each solution are calculated. Thus, we have Fscore(0), Fscore(1), Fscore(2), Fscore(3) and Fscore(4) corresponding to (W^0, M^0), (W^1, M^1), (W^2, M^2), (W^3, M^3) and (W^4, M^4), respectively. Assume Fscore(0) = 0.7, Fscore(1) = 0.8 , Fscore(2) = 0.7, Fscore(3) = 0.4 and Fscore(4) = 0.2 and let *minFinalSetSize* = 2.

Let *newSet* be an emptyset and *threshold* =1. While *newSet* is empty and *threshold* ≥ 1, the algorithm seeks a solution whose Fscore ≥ 1. Since there is no such solution, *newSet* remains empty and *threshold* is set to 0.9. Now, the algorithm seeks a solution whose Fscore ≥ 0.9. However, again, there is no such solution, thus *newSet* still remains empty and *threshold* is set to 0.8. Now the algorithm seeks for a solution with an Fscore ≥ 0.8. Eventually, the algorithm finds a solution satisfying the given condition. Then, the solution whose Fscore is 0.8 is added to the *newSet*. i.e. *newSet* = $\{(W^1, M^1)\}$, $\mathcal{X} \leftarrow \emptyset$,and $\mathcal{X} \leftarrow \{(W^1, M^1)\}$.

Now *newSet* is no longer empty, but since $|\mathcal{X}| <$ *minFinalSetSize*, new solutions should be generated by genetic operators. Note that $\mathcal{X} = \{(W^1, M^1)\}$. Then the algorithm uses $(W^1, M^1) \in \mathcal{X}$ to generate offspring by performing genetic operators.

Let new solutions be (W^5, M^5), (W^6, M^6), (W^7, M^7) and (W^8, M^8). Then $\mathcal{X} = \{(W^1, M^1), (W^5, M^5), (W^6, M^6), (W^7, M^7), (W^8, M^8)\}$. For each $(W^i, M^i) \in \mathcal{X}$ and for patients in $\mathcal{V}$, $U(p)$ and $\widetilde{U}(p)$ values are calculated to classify patients in the validation set. Then, due to the classification performance of the model in the validation set, sensitivity, specificity and Fscore values of each solution are calculated. Thus we have Fscore(1), Fscore(5), Fscore(6), Fscore(7) and Fscore(8) corresponding to (W^1, M^1), (W^5, M^5), (W^6, M^6), (W^7, M^7) and (W^8, M^8), respectively. Assume Fscore(1) = 0.8, Fscore(5) = 0.9 , Fscore(6) = 0.9 , Fscore(7) = 0.8 and Fscore(8) = 0.9.

Let *newSet* be an emptyset and *threshold* =1. While *newSet* is empty and *threshold* ≥ 1, the algorithm seeks a solution whose Fscore ≥ 1. Since there is no such solution, *newSet* remains empty and *threshold* is set to 0.9. Now, the algorithm seeks a solution with an Fscore ≥ 0.9. Since a solution that satisfies the given condition exists, this solution is added to *newSet*. i.e. *newSet* = $\{(W^5, M^5), (W^6, M^6), (W^8, M^8)\}$, $\mathcal{X} \leftarrow \emptyset$, and
$\mathcal{X} \leftarrow \{(W^5, M^5), (W^6, M^6), (W^8, M^8)\}$.

Now *newSet* is no longer empty, and $|\mathcal{X}| \geq$ *minFinalSetSize*. Thus final set of solutions that will be used for classifying observations in $\widetilde{\mathcal{S}}$ are obtained. Let p_{test_1} and p_{test_2} be the patients in test set, $\widetilde{\mathcal{S}}$. For each p $\in \widetilde{\mathcal{S}}$, the initial values of counters for positive and negative classification are zero. i.e. $CP(p_{test_1}) = 0$, $CN(p_{test_1}) = 0$, $CP(p_{test_2}) = 0$ and $CN(p_{test_2}) = 0$.

For each $(W^i, M^i) \in \mathcal{X}$ and for each patient in $\widetilde{\mathcal{S}}$, $U(p)$ and $\widetilde{U}(p)$ values are calculated. That is, by applying *GenerateUtilityFunctions*, $U(p_{test_1})$, $\widetilde{U}(p_{test_1})$, $U(p_{test_2})$ and $\widetilde{U}(p_{test_2})$ are calculated for each solution.

Assume that for (W^5, M^5), the global utilities for the patients in test set are as follows: $U(p_{test_1})$= 0.7, $\widetilde{U}(p_{test_1})$=0.75 and $U(p_{test_2})$=0.4, $\widetilde{U}(p_{test_2})$=0.8. Since $\widetilde{U}(p_{test_1}) \geq U(p_{test_1})$ and $\widetilde{U}(p_{test_2}) \geq U(p_{test_2})$, the classification of both p_{test_1} and p_{test_2} are determined as negative by this solution. That is, $CN(p_{test_1}) = 1$, $CP(p_{test_1}) = 0$, $CN(p_{test_2}) = 1$ and $CP(p_{test_2}) = 0$. The same procedure is repeated for other solutions in $\mathcal{X}$. Assume, for (W^6, M^6), patients p_{test_1} and p_{test_2} are classified as positive and negative respectively. In that case, the counters take the following values: $CN(p_{test_1}) = 1$, $CP(p_{test_1}) = 1$, $CN(p_{test_2}) = 2$ and $CP(p_{test_2}) = 0$. Finally, for the solution (W^8, M^8), both patients are classified as negative. That is $CN(p_{test_1}) = 2$, $CP(p_{test_1}) = 1$, $CN(p_{test_2}) = 3$ and $CP(p_{test_2}) = 0$.

The final classification decision is performed by comparing the counter values of a solution. That is, since $CN(p_{test_1}) = 2$ and $CP(p_{test_1}) = 1$, in consistent with majority voting, p_{test_1} is classified as negative. The final class decision of p_{test_2} is determined in a similar manner. That is, since $CN(p_{test_2}) = 3$ and $CP(p_{test_2}) = 0$, p_{test_2} is classified as negative.

B Hyper-parameter Optimization Procedure

B.1 Nested Cross Validation

To find the optimal values of the hyper-parameters, we applied hyper-parameter optimization. To do so, we run the model with different combinations of the values of hyper-parameters, and select the best combination according to model's classification performances. To determine the hyper-parameter values by an unbiased estimate of the generalization performance of the model, we apply 5-fold nested cross validation [96], where it is a common approach for hyper-parameter optimization [97, 98, 99].

It is defined as two nested loops of cross validation. Due to the inner cross validation performance, the hyper-parameter values are set and the outer loop evaluates the generalization ability of the model with the selected values of hyper-parameters on an independent set of observations [96, 97]. By this way, nested cross validation ensures that, the model do not use the observations reserved for outer loop to tune the hyper-parameters.

Once we divide the data into five folds, one fold is reserved for test and one fold of the remaining four folds are reserved for validation. For each combination of hyper-parameters, the model is trained with the remaining three folds and evaluated on the validation fold. This procedure should be repeated four times by rotating the validation fold among training folds. Thus, for a reserved test fold, and for a combination of hyper-parameters, there are four evaluations (i.e. Fscore). Then, the average of these four Fscore values of the inner loop are reported. This procedure is repeated five times, by having each fold as the test fold. The optimal hyper-parameter combination is the one whose reported average inner Fscore values are promising for all choices of the reserved test fold.

A set of hyper-parameter values are preferable if their inner loop performances are among the top one for all reserved test folds. However, if there is no such hyper-parameter set, we look for the one whose inner loop performances are among the top two, top three, top four etc. If there are more than one hyper-parameter set that satisfy the required conditions, we make the selection based on the higher average values or

lower standard deviations.

B.2 Hyper-parameter Optimization for PCM+NSGA-II

Table B.1 lists the potential values for the hyper-parameters of PCM+NSGA-II, which selected in a way that each hyper-parameter takes relatively low and high values.

Table B.1: Potential Values of Hyper-parameters of PCM+NSGA-II

PopulationSize	150, 250, 500, 1000
GenerationSize	5, 50, 100, 200
	s.t. *PopulationSize* $\geq$ *GenerationSize*
NumberOfGenerations	5, 10, 50, 100
p_{rc}, p_{lc}	(0.5, 0.5), (0.6, 0.4), (0.8, 0.2), (1, 0), (0.4, 0.6) ,(0.2, 0.8), (0,1)
	s.t. $p_{rc} + p_{lc} = 1$
p_m	0.01, 0.05, 0.1, 0.5

Under the given number of hyper-parameters and the given potential values, there are 1344 different combinations. Since it is computationally too expensive, we set their optimal values hierarchically. We first determine the values for *PopulationSize*, *GenerationSize* and *NumberOfGenerations* while fixing the values of $p_{rc} = 0.5$, $p_{lc} = 0.5$ and $p_m = 0.01$. Once the best values for these hyper-parameters are decided, we repeat the procedure to find the best value for p_{rc} and p_{lc} pair. Finally, we tune the p_m.

We apply 5-fold nested cross validation, and use Fscore to evaluate the performances of the different hyper-parameter choices.

B.3 Hyper-parameter Optimization for PCM+RECGA

Table B.2 lists the potential values for the hyper-parameters of PCM+RECGA, which selected in a way that each hyper-parameter takes relatively low and high values.

Table B.2: Potential Values of Hyper-parameters of PCM+RECGA

PopulationSize	150, 250, 500, 1000
minFinalSetSize	5, 50, 100, 200
	s.t. *PopulationSize* $\geq$ *minFinalSetSize*
p_{rc}, p_{lc}	(0.5, 0.5), (0.6, 0.4), (0.8, 0.2), (1, 0), (0.4, 0.6) ,(0.2, 0.8), (0,1)
	s.t. $p_{rc} + p_{lc} = 1$
p_m	0.01, 0.05, 0.1, 0.5

Under the given number of hyper-parameters and the given potential values, there are 420 different combinations. Since it is computationally too expensive, we set their optimal values hierarchically. We first determine the values for ***PopulationSize***, ***GenerationSize*** and ***NumberOfGenerations*** while fixing the values of $p_{rc} = 0.5$, $p_{lc} = 0.5$ and $p_m = 0.01$. Once the best values for these hyper-parameters are decided, we repeat the procedure to find the best value for p_{rc} and p_{lc} pair. Finally, we tune the p_m.

We apply 5-fold nested cross validation, and use Fscore to evaluate the performances of the different hyper-parameter value choices.

C Hyper-parameter Optimization for the In-Stent-Restenosis Dataset

For the experiments conducted with in-stent-restenosis dataset, we created two different settings. Settings are differentiated by the ratio of the number of patients with and without restenosis in training, validation and test samples. The corresponding folds are created as given in Table C.1.

Table C.1: Content of a Fold for Setting 1 and Setting 2

	Setting 1	**Setting 2**
# of patients with restenosis	12	12
# of patients without restenosis	48	12

C.1 Hyper-parameter Optimization for PCM+NSGA-II

C.1.1 Setting 1

Table C.2 indicates the Fscore performances of inner cross validation for different values of *PopulationSize*, *GenerationSize* and *NumberOfGenerations*.

There is no hyper-parameter set whose inner cross validation performances are among the top one, top two or top three for all repetition, thus, the hyper-parameter values whose inner cross validation performances are among the top four for all the given test folds are marked with **** in the last column.

Thus, the *PopulationSize*, *GenerationSize* and *NumberOfGenerations* are set to 1000, 50 and 5, respectively. Once these values are set, we repeat the same analysis for p_{rc} and p_{lc}.

The hyper-parameter values whose inner cross validation performances are among the top three for all the given test folds are marked with *** in the last column of Table C.3. Their average Fscore values are also indicated in parenthesis. Since they are equal, we prefer to set $p_{rc} = 0.5$ and $p_{lc} = 0.5$ due to their lower standard deviation among five runs.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top two for all the given test folds are marked with ** in the last column of Table C.4. Thus we set p_m=0.01.

Table C.2: Inner Loop Performances for *PopulationSize*, *GenerationSize* and *NumberOfGenerations* (PCM+NSGA-II, Setting 1)

PopulationSize	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	**5**	0.62	0.71	0.71	0.69	0.73	**250**	**100**	**5**	0.72	0.71	0.70	0.67	0.74	**500**	**200**	**5**	0.72	0.72	0.69	0.67	0.74	
150	**5**	**10**	0.52	0.61	0.65	0.63	0.67	**250**	**100**	**10**	0.67	0.72	0.73	0.71	0.76	**500**	**200**	**10**	0.74	0.73	0.66	0.65	0.71	
150	**5**	**50**	0.54	0.68	0.66	0.64	0.67	**250**	**100**	**50**	0.65	0.65	0.68	0.64	0.70	**500**	**200**	**50**	0.65	0.71	0.66	0.65	0.71	
150	**5**	**100**	0.51	0.62	0.63	0.62	0.64	**250**	**100**	**100**	0.67	0.73	0.73	0.70	0.76	**500**	**200**	**100**	0.67	0.71	0.66	0.65	0.71	
150	**50**	**5**	0.73	0.72	0.71	0.68	0.74	**250**	**200**	**5**	0.64	0.69	0.67	0.65	0.70	**1000**	**5**	**5**	0.64	0.68	0.68	0.67	0.70	
150	**50**	**10**	0.71	0.73	0.71	0.69	0.74	**250**	**200**	**10**	0.69	0.70	0.67	0.66	0.71	**1000**	**5**	**10**	0.56	0.52	0.53	0.52	0.56	
150	**50**	**50**	0.68	0.75	0.70	0.68	0.73	**250**	**200**	**50**	0.68	0.64	0.62	0.59	0.67	**1000**	**5**	**50**	0.66	0.58	0.52	0.52	0.59	
150	**50**	**100**	0.69	0.73	0.69	0.67	0.72	**250**	**200**	**100**	0.63	0.66	0.68	0.65	0.71	**1000**	**5**	**100**	0.72	0.68	0.70	0.69	0.72	
150	**100**	**5**	0.67	0.69	0.68	0.65	0.70	**500**	**5**	**5**	0.59	0.62	0.59	0.55	0.59	**1000**	**50**	**5**	**0.75**	**0.71**	**0.71**	**0.69**	**0.74**	****(0.72)**
150	**100**	**10**	0.72	0.71	0.71	0.68	0.73	**500**	**5**	**10**	0.62	0.61	0.65	0.60	0.65	**1000**	**50**	**10**	0.71	0.67	0.71	0.66	0.72	
150	**100**	**50**	0.71	0.72	0.67	0.67	0.72	**500**	**5**	**50**	0.65	0.66	0.61	0.60	0.64	**1000**	**50**	**50**	0.64	0.69	0.65	0.66	0.69	
150	**100**	**100**	0.71	0.71	0.70	0.70	0.74	**500**	**5**	**100**	0.76	0.68	0.67	0.65	0.69	**1000**	**50**	**100**	0.59	0.67	0.70	0.66	0.71	
250	**5**	**5**	0.63	0.60	0.57	0.55	0.60	**500**	**50**	**5**	0.74	0.73	0.71	0.68	0.74	**1000**	**100**	**5**	0.71	0.73	0.74	0.72	0.76	
250	**5**	**10**	0.67	0.66	0.64	0.60	0.67	**500**	**50**	**10**	0.71	0.73	0.71	0.69	0.74	**1000**	**100**	**10**	0.69	0.72	0.70	0.67	0.73	
250	**5**	**50**	0.72	0.72	0.67	0.66	0.70	**500**	**50**	**50**	0.71	0.73	0.71	0.68	0.74	**1000**	**100**	**50**	0.66	0.69	0.73	0.69	0.74	
250	**5**	**100**	0.61	0.60	0.61	0.60	0.64	**500**	**50**	**100**	0.71	0.72	0.68	0.67	0.72	**1,000**	**100**	**100**	0.62	0.72	0.73	0.71	0.75	
250	**50**	**5**	0.72	0.71	0.70	0.68	0.73	**500**	**100**	**5**	0.70	0.71	0.70	0.68	0.74	**1000**	**200**	**5**	0.71	0.73	0.70	0.67	0.74	
250	**50**	**10**	0.71	0.71	0.70	0.68	0.73	**500**	**100**	**10**	0.71	0.73	0.70	0.67	0.73	**1000**	**200**	**10**	0.72	0.73	0.70	0.67	0.74	
250	**50**	**50**	0.65	0.69	0.73	0.68	0.74	**500**	**100**	**50**	0.65	0.68	0.72	0.67	0.74	**1000**	**200**	**50**	0.69	0.70	0.72	0.70	0.75	
250	**50**	**100**	0.63	0.66	0.67	0.64	0.69	**500**	**100**	**100**	0.71	0.69	0.67	0.64	0.70	**1000**	**200**	**100**	0.68	0.68	0.72	0.67	0.73	

Table C.3: Inner Loop Performances for p_{rc}, p_{lc} (PCM+NSGA-II, Setting 1)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	**0.75**	**0.71**	**0.71**	**0.69**	**0.74**	***(0.72)**
0.6-0.4	0.74	0.70	0.69	0.68	0.73	
0.8-0.2	0.68	0.72	0.72	0.69	0.75	
1.0-0.0	0.72	0.71	0.72	0.69	0.74	
0.4-0.6	0.74	0.69	0.65	0.64	0.69	
0.2-0.8	0.73	0.74	0.71	0.68	0.74	***(0.72)
0.0-1.0	0.68	0.70	0.71	0.66	0.72	

Table C.4: Inner Loop Performances for p_m (PCM+NSGA-II, Setting 1)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.75**	**0.71**	**0.71**	**0.69**	**0.74**	**(0.72)**
0.05	0.69	0.71	0.71	0.68	0.74	
0.10	0.71	0.71	0.66	0.65	0.72	
0.50	0.70	0.73	0.71	0.69	0.74	

C.1.2 Setting 2

Table C.5 indicates the Fscore performances of inner cross validation for different values of *PopulationSize*, *GenerationSize* and *NumberOfGenerations*.

The hyper-parameter values whose inner cross validation performances are among the top three for all the given test folds are marked with *** in the last column. Due to its higher average value, we select the hyper-parameter set of 1000, 100, 100 for *PopulationSize*, *GenerationSize* and *NumberOfGenerations*.

Then we repeat the same analysis for p_{rc} and p_{lc}.

The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column of Table C.6. We set $p_{rc} = 0.5$ and $p_{lc} = 0.5$.

Finally, we tune the value of p_m by following the same procedure.

The hyper-parameter value whose inner cross validation performance is among the top two for all the given test folds are marked with ** in the last column of Table C.7. Thus we set p_m=0.01.

Table C.5: Inner Loop Performances for *PopulationSize*, *GenerationSize* and *NumberOfGenerations* (PCM+NSGA-II, Setting 2)

PopulationSize	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5		**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	**5**	0.67	0.64	0.70	0.67	0.72	**250**	**100**	**5**	0.66	0.65	0.67	0.69	0.71		**500**	**200**	**5**	0.62	0.62	0.65	0.66	0.68	
150	**5**	**10**	0.62	0.64	0.69	0.66	0.71	**250**	**100**	**10**	0.72	0.64	0.70	0.66	0.71		**500**	**200**	**10**	0.71	0.65	0.71	0.67	0.71	
150	**5**	**50**	0.61	0.64	0.70	0.67	0.72	**250**	**100**	**50**	0.72	0.68	0.72	0.70	0.73		**500**	**200**	**50**	0.70	0.67	0.69	0.67	0.71	
150	**5**	**100**	0.62	0.65	0.70	0.67	0.72	**250**	**100**	**100**	0.68	0.69	0.72	0.70	0.73		**500**	**200**	**100**	0.68	0.68	0.71	0.69	0.72	
150	**50**	**5**	0.65	0.68	0.73	0.71	0.75	**250**	**200**	**5**	0.66	0.61	0.65	0.65	0.68		**1000**	**5**	**5**	0.63	0.65	0.70	0.67	0.70	
150	**50**	**10**	0.68	0.67	0.74	0.72	0.75	**250**	**200**	**10**	0.66	0.62	0.65	0.66	0.68		**1000**	**5**	**10**	0.63	0.61	0.68	0.64	0.69	
150	**50**	**50**	0.69	0.68	0.73	0.72	0.74	**250**	**200**	**50**	0.70	0.69	0.72	0.70	0.73		**1000**	**5**	**50**	0.63	0.62	0.71	0.68	0.71	
150	**50**	**100**	0.69	0.67	0.74	0.73	0.75	**250**	**200**	**100**	0.70	0.64	0.72	0.67	0.71		**1000**	**5**	**100**	0.62	0.61	0.65	0.63	0.67	
150	**100**	**5**	0.64	0.59	0.62	0.61	0.66	**500**	**5**	**5**	0.47	0.61	0.71	0.67	0.71		**1000**	**50**	**5**	0.70	0.67	0.73	0.71	0.75	
150	**100**	**10**	0.65	0.63	0.71	0.71	0.72	**500**	**5**	**10**	0.62	0.67	0.71	0.67	0.71		**1000**	**50**	**10**	0.64	0.67	0.75	0.73	0.75	
150	**100**	**50**	0.70	0.68	0.75	0.73	0.75	**500**	**5**	**50**	0.62	0.64	0.68	0.67	0.69		**1000**	**50**	**50**	0.68	0.62	0.71	0.65	0.69	
150	**100**	**100**	0.68	0.66	0.74	0.72	0.74	**500**	**5**	**100**	0.62	0.65	0.70	0.67	0.70		**1000**	**50**	**100**	0.70	0.63	0.69	0.67	0.69	
250	**5**	**5**	0.65	0.66	0.70	0.70	0.70	**500**	**50**	**5**	0.72	0.69	0.75	0.73	0.75	***(0.728)	**1000**	**100**	**5**	0.71	0.69	0.75	0.73	0.76	
250	**5**	**10**	0.60	0.64	0.72	0.70	0.72	**500**	**50**	**10**	0.68	0.67	0.75	0.73	0.75		**1000**	**100**	**10**	0.71	0.68	0.74	0.72	0.74	
250	**5**	**50**	0.63	0.65	0.67	0.67	0.68	**500**	**50**	**50**	0.72	0.67	0.74	0.73	0.75	***(0.722)	**1000**	**100**	**50**	0.68	0.66	0.71	0.67	0.70	
250	**5**	**100**	0.60	0.63	0.68	0.68	0.69	**500**	**50**	**100**	0.64	0.66	0.73	0.73	0.74		**1000**	**100**	**100**	**0.72**	**0.69**	**0.75**	**0.73**	**0.76**	***(0.730)**
250	**50**	**5**	0.71	0.68	0.75	0.73	0.76	**500**	**100**	**5**	0.70	0.65	0.72	0.67	0.71		**1000**	**200**	**5**	0.63	0.61	0.65	0.65	0.67	
250	**50**	**10**	0.68	0.67	0.74	0.72	0.74	**500**	**100**	**10**	0.73	0.64	0.72	0.67	0.71		**1000**	**200**	**10**	0.70	0.62	0.65	0.66	0.68	
250	**50**	**50**	0.67	0.67	0.74	0.72	0.75	**500**	**100**	**50**	0.73	0.64	0.70	0.66	0.70		**1000**	**200**	**50**	0.70	0.64	0.69	0.67	0.71	
250	**50**	**100**	0.71	0.67	0.75	0.73	0.76	**500**	**100**	**100**	0.68	0.65	0.72	0.69	0.72		**1000**	**200**	**100**	0.74	0.63	0.69	0.65	0.70	

Table C.6: Inner Loop Performances for p_{rc}, p_{lc} (PCM+NSGA-II, Setting 2)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	**0.72**	**0.69**	**0.75**	**0.73**	**0.76**	****(0.73)**
0.6-0.4	0.71	0.63	0.69	0.65	0.70	
0.8-0.2	0.69	0.68	0.72	0.70	0.73	
1.0-0.0	0.62	0.70	0.72	0.72	0.75	
0.4-0.6	0.68	0.68	0.69	0.67	0.71	
0.2-0.8	0.71	0.64	0.70	0.68	0.70	
0.0-1.0	0.68	0.67	0.70	0.69	0.73	

Table C.7: Inner Loop Performances for p_m (PCM+NSGA-II, Setting 2)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.72**	**0.69**	**0.75**	**0.73**	**0.76**	****(0.73)**
0.05	0.74	0.64	0.65	0.61	0.67	
0.10	0.67	0.63	0.65	0.61	0.67	
0.50	0.66	0.63	0.68	0.66	0.68	

C.2 Hyper-parameter Optimization for PCM+RECGA

C.2.1 Setting 1

Table C.8 indicates Fscore performances of inner cross validation for different values of *PopulationSize* and *minFinalSetSize*.

The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column. We set

PopulationSize, and *NumberOfGenerations* as 250 and 100, respectively.

Then we repeat the same analysis for p_{rc} and p_{lc}. The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column of Table C.9. Here, we have an exceptional case. Even their inner cross validation performance are not in top one or top two for all the given test folds, due to their remarkably high performance on four out of five iterations of inner cross validation we set p_{rc} and p_{lc} as 1.0 and 0.0.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top two for all the given test folds are marked with ** in the last column of Table C.10. Thus we set p_m = 0.50.

Table C.8: Inner Loop Performances for *PopulationSize* and *minFinalSetSize* (PCM+RECGA, Setting 1)

PopulationSize	**minFinalSetSize**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	0.71	0.69	0.67	0.65	0.69	
150	**50**	0.71	0.69	0.67	0.65	0.69	
150	**100**	0.71	0.69	0.67	0.65	0.69	
250	**5**	0.69	0.69	0.67	0.65	0.69	
250	**50**	0.69	0.69	0.67	0.65	0.69	
250	**100**	**0.70**	**0.69**	**0.67**	**0.65**	**0.69**	****(0.68)**
250	**200**	0.70	0.69	0.67	0.65	0.69	**(0.68)
500	**5**	0.67	0.69	0.67	0.65	0.69	
500	**50**	0.67	0.69	0.67	0.65	0.69	
500	**100**	0.67	0.69	0.67	0.65	0.69	
500	**200**	0.69	0.70	0.67	0.65	0.69	
1000	**5**	0.67	0.69	0.67	0.65	0.69	
1000	**50**	0.67	0.69	0.67	0.65	0.69	
1000	**100**	0.67	0.69	0.67	0.65	0.69	
1000	**200**	0.68	0.69	0.67	0.65	0.69	

Table C.9: Inner Loop Performances for p_{rc}, p_{lc} (PCM+RECGA, Setting 1)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	0.70	0.69	0.67	0.65	0.69	**(0.68)
0.6-0.4	0.69	0.68	0.67	0.64	0.68	
0.8-0.2	0.66	0.69	0.67	0.65	0.69	
1.0-0.0	**0.66**	**0.71**	**0.72**	**0.70**	**0.75**	**!(0.71)**
0.4-0.6	0.70	0.68	0.67	0.65	0.69	
0.2-0.8	0.69	0.69	0.67	0.65	0.69	
0.0-1.0	0.72	0.66	0.67	0.64	0.68	

Table C.10: Inner Loop Performances for p_m (PCM+RECGA, Setting 1)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	0.66	0.71	0.72	0.70	0.75	
0.05	0.66	0.69	0.70	0.70	0.73	
0.10	0.70	0.69	0.66	0.64	0.69	
0.50	**0.71**	**0.70**	**0.70**	**0.69**	**0.75**	**(0.71)**

C.2.2 Setting 2

Table C.11 indicates the Fscore performances of inner cross validation for different values of *PopulationSize* and *minFinalSetSize*.

The hyper-parameter values whose inner cross validation performances are among the top one for all the given test folds are marked with * in the last column. We set *PopulationSize*, and *NumberOfGenerations* as 150 and 50, respectively.

Then we repeat the same analysis for p_{rc} and p_{lc}. The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column of Table C.12 and we set p_{rc} and p_{lc} as 0.5.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top one for all the given test folds are marked with * in the last column of Table C.13. Thus we set p_m = 0.01.

Table C.11: Inner Loop Performances for *PopulationSize* and *minFinalSetSize* (PCM+RECGA, Setting 2)

PopulationSize	**minFinalSetSize**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	0.70	0.68	0.70	0.70	0.73	
150	**50**	**0.74**	**0.68**	**0.70**	**0.70**	**0.73**	***(0.71)**
150	**100**	0.74	0.68	0.70	0.70	0.73	*(0.71)
250	**5**	0.71	0.68	0.70	0.70	0.73	
250	**50**	0.70	0.68	0.70	0.70	0.73	
250	**100**	0.70	0.68	0.70	0.70	0.73	
250	**200**	0.70	0.68	0.70	0.70	0.73	
500	**5**	0.65	0.67	0.69	0.70	0.73	
500	**50**	0.73	0.67	0.69	0.70	0.73	
500	**100**	0.73	0.67	0.69	0.70	0.73	
500	**200**	0.73	0.67	0.69	0.70	0.73	
1000	**5**	0.64	0.68	0.69	0.70	0.73	
1000	**50**	0.71	0.68	0.69	0.70	0.73	
1000	**100**	0.71	0.68	0.69	0.70	0.73	
1000	**200**	0.71	0.68	0.69	0.70	0.73	

Table C.12: Inner Loop Performances for p_{rc}, p_{lc} (PCM+RECGA, Setting 2)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	**0.74**	**0.68**	**0.70**	**0.70**	**0.73**	****(0.71)**
0.6-0.4	0.74	0.68	0.70	0.70	0.73	
0.8-0.2	0.57	0.63	0.66	0.66	0.69	
1.0-0.0	0.62	0.63	0.66	0.66	0.69	
0.4-0.6	0.67	0.68	0.70	0.70	0.73	**(0.69)
0.2-0.8	0.66	0.69	0.70	0.70	0.73	
0.0-1.0	0.63	0.68	0.70	0.70	0.73	

Table C.13: Inner Loop Performances for p_m (PCM+RECGA, Setting 2)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.74**	**0.68**	**0.70**	**0.70**	**0.73**	***(0.71)**
0.05	0.64	0.63	0.66	0.66	0.69	
0.10	0.61	0.65	0.63	0.65	0.69	
0.50	0.65	0.63	0.70	0.65	0.69	

D Generalization Performances of the Models for the In-Stent-Restenosis Dataset: 5-fold Cross Validation

Recall that, in each iteration of inner cross validation, a fold is reserved for test, which is not used in the hyper-parameter tuning process. To see the generalization performance of the models with the selected hyper-parameter values, we test the model in outer loop of the nested cross validation. Thus, we test the model performances with 5-fold cross validation.

The details of the settings are given in Table D.1. The table indicates that, in Setting 1, for each run of cross validation, there are 24 positive and 96 negative observations in $\mathcal{S}$ and $\mathcal{V}$, and there are 12 positive and 48 negative observations in $\widetilde{\mathcal{S}}$. In a similar manner, in Setting 2, for each run of cross validation, there are 24 positive and 24 negative observations in $\mathcal{S}$ and $\mathcal{V}$, and the $\widetilde{\mathcal{S}}$ is comprised of 12 positive and 12 negative observations.

The last row of Table D.1 indicates the interval of values that the parameter L of PCM can take.

Table D.1: In-Stent-Restenosis Dataset, Settings

	Setting 1			**Setting 2**		
	$\mathcal{S}$	$\mathcal{V}$	$\widetilde{\mathcal{S}}$	$\mathcal{S}$	$\mathcal{V}$	$\widetilde{\mathcal{S}}$
# of patients with restenosis	24	24	12	24	24	12
# of patients without restenosis	96	96	48	24	24	12
Total # of patients	120	120	60	48	48	24
L	$\{0, \ldots, 96\}$			$\{0, \ldots, 24\}$		

Other than PCM+NSGA-II and PCM+RECGA, we also repeat the experiments of 5-fold cross validation for the competitor models for the sake of completeness. Note that, PCM+NSGA-II and PCM+RECGA first utilize $\mathcal{S}$ to generate initial set of solutions, and then they tune these solutions with $\mathcal{V}$. On the other hand, competitor models use $\mathcal{S} \cup \mathcal{V}$ for training. All the models' performances are tested in $\widetilde{\mathcal{S}}$.

Tables D.2 and D.3, D.4 and D.4 summarize the 5-fold CV performances of the models on in-stent-restenosis dataset.

Table D.2: Average Training Performance Results (5-fold CV)

	PCM+NSGA-II	PCM+RECGA	LR	pen-LR	SVM	ANN	DT	RF
				Setting 1				
Sensitivity	0.65	0.77	0.34	0.34	0.20	0.40	0.45	0.55
Specificity	0.85	0.70	0.98	0.98	1.00	0.98	0.96	0.97
Accuracy	0.81	0.71	0.85	0.85	0.84	0.86	0.86	0.89
Fscore	0.73	0.72	0.50	0.50	0.29	0.56	0.61	0.70
Fmeasure	0.58	0.52	0.47	0.47	0.29	0.53	0.56	0.66
				Setting 2				
Sensitivity	0.72	0.77	0.77	0.77	0.85	0.85	0.77	0.92
Specificity	0.72	0.64	0.70	0.70	0.64	0.76	0.83	0.81
Accuracy	0.72	0.70	0.74	0.74	0.75	0.81	0.80	0.87
Fscore	0.71	0.70	0.73	0.73	0.73	0.79	0.80	0.86
Fmeasure	0.72	0.72	0.75	0.75	0.77	0.81	0.79	0.87

Table D.3: Standard Deviations of Training Performance Indicators (5-fold CV)

	PCM+NSGA-II	PCM+RECGA	LR	pen-LR	SVM	ANN	DT	RF
				Setting 1				
Sensitivity	0.06	0.09	0.03	0.03	0.16	0.10	0.07	0.04
Specificity	0.06	0.09	0.01	0.01	0.00	0.02	0.01	0.01
Accuracy	0.05	0.06	0.01	0.01	0.03	0.01	0.01	0.01
Fscore	0.04	0.03	0.04	0.04	0.24	0.09	0.07	0.03
Fmeasure	0.08	0.04	0.03	0.03	0.24	0.07	0.05	0.03
				Setting 2				
Sensitivity	0.04	0.03	0.03	0.03	0.05	0.13	0.03	0.04
Specificity	0.05	0.03	0.05	0.05	0.09	0.08	0.02	0.05
Accuracy	0.01	0.02	0.03	0.03	0.05	0.03	0.01	0.02
Fscore	0.01	0.02	0.03	0.03	0.06	0.02	0.01	0.02
Fmeasure	0.01	0.02	0.02	0.02	0.04	0.04	0.01	0.01

Table D.4: Average Test Performance Results (5-fold CV)

	PCM+NSGA-II	PCM+RECGA	LR	pen-LR	SVM	ANN	DT	RF
Setting 1								
Sensitivity	0.58	0.65	0.33	0.33	0.10	0.25	0.33	0.30
Specificity	0.83	0.69	0.98	0.98	0.98	0.95	0.93	0.92
Accuracy	0.78	0.68	0.85	0.85	0.81	0.81	0.81	0.79
Fscore	0.67	0.65	0.49	0.49	0.17	0.39	0.48	0.44
Fmeasure	0.51	0.45	0.46	0.46	0.16	0.35	0.40	0.36
Setting 2								
Sensitivity	0.62	0.72	0.75	0.73	0.80	0.77	0.63	0.77
Specificity	0.73	0.65	0.72	0.72	0.53	0.62	0.73	0.67
Accuracy	0.68	0.68	0.73	0.73	0.67	0.69	0.68	0.72
Fscore	0.65	0.67	0.72	0.71	0.63	0.62	0.67	0.70
Fmeasure	0.64	0.69	0.73	0.72	0.70	0.71	0.66	0.73

Table D.5: Standard Deviations of Test Performance Indicators (5-fold CV)

	PCM+NSGA-II	PCM+RECGA	LR	pen-LR	SVM	ANN	DT	RF
Setting 1								
Sensitivity	0.17	0.14	0.07	0.07	0.08	0.05	0.12	0.08
Specificity	0.04	0.10	0.02	0.02	0.02	0.02	0.04	0.04
Accuracy	0.05	0.06	0.02	0.02	0.01	0.02	0.04	0.03
Fscore	0.12	0.04	0.08	0.08	0.14	0.07	0.13	0.10
Fmeasure	0.12	0.05	0.09	0.09	0.13	0.06	0.13	0.08
Setting 2								
Sensitivity	0.17	0.15	0.14	0.15	0.11	0.18	0.12	0.10
Specificity	0.10	0.06	0.11	0.11	0.12	0.23	0.08	0.12
Accuracy	0.07	0.06	0.08	0.09	0.07	0.04	0.06	0.04
Fscore	0.08	0.05	0.07	0.08	0.08	0.12	0.07	0.05
Fmeasure	0.10	0.08	0.09	0.10	0.07	0.05	0.08	0.04

Note that the given performances of PCM+NSGA-II and PCM+RECGA under the experimental setting with 5-fold cross validation are similar to the performances when the experimental analysis conducted with randomly generated 100 instances.

E Hyper-parameter Optimization for the Wisconsin Breast Cancer Original Dataset

Table E.1 summarizes the content of a fold. Note that, the positive and negative observations in a fold are arranged to satisfy predetermined rareness levels. Columns "Malign" and "Benign" of Table E.1 indicate the number of positive and negative observations in a fold and "Rareness level" indicates the ratio of positive observations to all observations.

Table E.1: Content of a Fold for Different Rareness Levels- WBCO Dataset

		Malign	Benign
Rareness level	1%	1	88
	3%	3	88
	5%	5	88
	7%	7	88
	10%	10	88
	15%	16	88
	25%	30	88
	35%	47	88

E.1 Hyper-parameter Optimization for PCM+NSGA-II

E.1.1 Rareness Level = 1%

Table E.2 indicates the Fscore performances of inner cross validation for different values of *PopulationSize*, *GenerationSize* and *NumberOfGenerations*.

The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column. Thus, the *PopulationSize*, *GenerationSize* and *NumberOfGenerations* are set to 1000, 50 and 5, respectively. Once these values are set, we repeat the same analysis for p_{rc} and p_{lc}. The hyper-parameter values whose inner cross validation performances are

among the top one for all the given test folds are marked with * in the last column of Table E.3. Their average Fscore values are also indicated in parenthesis. Since they are equal, we set $p_{rc} = 0.5$ and $p_{lc} = 0.5$.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top one for all the given test folds are marked with * in the last column of Table E.4. Thus we set p_m=0.01.

Table E.2: Inner Loop Performances for *PopulationSize*, *GenerationSize* and *NumberOfGenerations* (PCM+NSGA-II, Rareness Level=1%)

PopulationSize	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	**5**	0.74	0.50	0.50	0.75	0.75	**250**	**100**	**5**	0.89	0.95	0.97	0.97	0.97	**500**	**200**	**5**	0.86	0.86	0.87	0.87	0.87	
150	**5**	**10**	0.74	0.50	0.50	0.75	0.75	**250**	**100**	**10**	0.90	0.95	0.97	0.97	0.97	**500**	**200**	**10**	0.87	0.81	0.82	0.83	0.82	
150	**5**	**50**	0.75	0.50	0.50	0.75	0.75	**250**	**100**	**50**	0.89	0.95	0.96	0.97	0.97	**500**	**200**	**50**	0.92	0.73	0.75	0.99	0.75	
150	**5**	**100**	0.50	0.50	0.50	0.75	0.75	**250**	**100**	**100**	0.94	0.97	0.96	0.97	0.97	**500**	**200**	**100**	0.87	0.73	0.74	0.99	0.74	
150	**50**	**5**	0.97	0.99	0.98	0.98	0.98	**250**	**200**	**5**	0.80	0.84	0.87	0.87	0.86	**1000**	**5**	**5**	0.50	0.50	0.50	0.75	0.75	
150	**50**	**10**	0.97	0.74	0.74	0.98	0.74	**250**	**200**	**10**	0.80	0.81	0.83	0.83	0.83	**1000**	**5**	**10**	0.50	0.50	0.50	0.75	0.75	
150	**50**	**50**	0.70	0.74	0.49	0.74	0.49	**250**	**200**	**50**	0.76	0.85	0.89	0.89	0.89	**1000**	**5**	**50**	0.25	0.75	0.50	0.75	0.50	
150	**50**	**100**	0.70	0.74	0.49	0.74	0.49	**250**	**200**	**100**	0.77	0.86	0.91	0.90	0.90	**1000**	**5**	**100**	0.50	0.50	0.50	0.75	0.75	
150	**100**	**5**	0.92	0.91	0.92	0.91	0.91	**500**	**5**	**5**	0.50	0.50	0.50	0.75	0.75	**1000**	**50**	**5**	**0.98**	**0.99**	**0.99**	**0.99**	**0.99**	****(0.99)**
150	**100**	**10**	0.94	0.95	0.95	0.95	0.95	**500**	**5**	**10**	0.75	0.50	0.50	0.75	0.75	**1000**	**50**	**10**	0.73	0.75	0.75	0.99	0.75	
150	**100**	**50**	0.93	0.98	0.97	0.97	0.97	**500**	**5**	**50**	0.50	0.74	0.75	1.00	0.75	**1000**	**50**	**50**	0.73	0.98	0.98	0.98	0.98	
150	**100**	**100**	0.93	0.97	0.97	0.97	0.97	**500**	**5**	**100**	0.50	0.74	0.74	0.99	0.75	**1000**	**50**	**100**	0.92	0.97	0.98	0.98	0.98	
250	**5**	**5**	0.50	0.50	0.50	0.75	0.75	**500**	**50**	**5**	0.97	0.74	0.74	0.74	0.99	**1000**	**100**	**5**	0.86	0.92	0.96	0.95	0.95	
250	**5**	**10**	0.50	0.50	0.50	0.75	0.75	**500**	**50**	**10**	0.96	0.74	0.74	0.99	0.74	**1000**	**100**	**10**	0.80	0.92	0.99	0.99	0.99	
250	**5**	**50**	0.50	0.50	0.50	0.75	0.75	**500**	**50**	**50**	0.95	0.97	0.98	0.98	0.98	**1000**	**100**	**50**	0.73	0.99	0.99	0.99	0.99	
250	**5**	**100**	0.75	0.50	0.50	0.75	0.75	**500**	**50**	**100**	0.95	0.98	0.98	0.98	0.98	**1,000**	**100**	**100**	0.90	0.72	0.49	0.74	0.49	
250	**50**	**5**	0.95	0.99	0.99	0.99	0.99	**500**	**100**	**5**	0.91	0.94	0.94	0.94	0.94	**1000**	**200**	**5**	0.89	0.95	0.97	0.97	0.96	
250	**50**	**10**	0.97	0.74	0.74	0.74	0.99	**500**	**100**	**10**	0.96	0.95	0.96	0.96	0.95	**1000**	**200**	**10**	0.98	0.74	0.74	0.74	0.99	
250	**50**	**50**	0.74	0.49	0.49	0.74	0.74	**500**	**100**	**50**	0.93	0.95	0.96	0.96	0.96	**1000**	**200**	**50**	0.81	0.44	0.49	0.74	0.74	
250	**50**	**100**	0.95	0.49	0.49	0.74	0.74	**500**	**100**	**100**	0.82	0.91	0.96	0.95	0.95	**1000**	**200**	**100**	0.79	0.67	0.74	0.99	0.74	

Table E.3: Inner Loop Performances for p_{rc}, p_{lc} (PCM+NSGA-II, Rareness Level=1%)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	**0.98**	**0.99**	**0.99**	**0.99**	**0.99**	***(0.99)**
0.6-0.4	0.97	0.49	0.50	0.74	0.74	
0.8-0.2	0.98	0.74	0.74	0.99	0.74	
1.0-0.0	0.98	0.99	0.99	0.99	0.99	*(0.99)
0.4-0.6	0.96	0.50	0.50	0.75	0.75	
0.2-0.8	0.74	0.99	0.99	0.99	0.99	
0.0-1.0	0.96	0.98	0.98	0.98	0.98	

Table E.4: Inner Loop Performances for p_m (PCM+NSGA-II, Rareness Level=1%)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.98**	**0.99**	**0.99**	**0.99**	**0.99**	***(0.99)**
0.05	0.97	0.74	0.74	0.99	0.75	
0.10	0.97	0.99	0.99	0.99	0.99	
0.50	0.95	0.49	0.49	0.74	0.74	

E.1.2 Rareness Level = 10%

Table E.5 indicates the Fscore performances of inner cross validation for different values of *PopulationSize*, *GenerationSize* and *NumberOfGenerations*.

The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column.

Thus, the *PopulationSize*, *GenerationSize* and *NumberOfGenerations* are set to 500, 5 and 5, respectively. Once these values are set, we repeat the same analysis for p_{rc} and p_{lc}.

The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column of Table E.6. Their average Fscore values are also indicated in parenthesis. Due to their higher average value, we set $p_{rc} = 0.5$ and $p_{lc} = 0.5$.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top one for all the given test folds are marked with * in the last column of Table E.7. Thus we set p_m=0.01.

Table E.5: Inner Loop Performances for *PopulationSize*, *GenerationSize* and *NumberOfGenerations* (PCM+NSGA-II, Rareness Level=10%)

PopulationSize	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5		**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5
150	**5**	**5**	0.96	0.96	0.97	0.97	0.97	**250**	**100**	**5**	0.99	0.96	0.94	0.95	0.94		**500**	**200**	**5**	0.97	0.97	0.95	0.95	0.95
150	**5**	**10**	0.97	0.96	0.99	0.99	0.99	**250**	**100**	**10**	0.96	0.96	0.94	0.95	0.94		**500**	**200**	**10**	0.94	0.97	0.96	0.97	0.96
150	**5**	**50**	0.98	0.96	0.98	0.98	0.98	**250**	**100**	**50**	0.92	0.95	0.88	0.91	0.89		**500**	**200**	**50**	0.96	0.97	0.95	0.95	0.95
150	**5**	**100**	0.98	0.91	0.93	0.93	0.90	**250**	**100**	**100**	0.93	0.96	0.88	0.91	0.88		**500**	**200**	**100**	0.91	0.98	0.95	0.95	0.95
150	**50**	**5**	0.99	0.98	0.95	0.95	0.95	**250**	**200**	**5**	0.99	0.97	0.95	0.95	0.94		**1000**	**5**	**5**	0.92	0.96	0.99	0.99	0.99
150	**50**	**10**	0.98	0.95	0.95	0.95	0.95	**250**	**200**	**10**	0.96	0.97	0.95	0.95	0.94		**1000**	**5**	**10**	0.93	0.96	0.98	0.98	0.98
150	**50**	**50**	0.97	0.93	0.95	0.94	0.94	**250**	**200**	**50**	0.91	0.98	0.94	0.94	0.93		**1000**	**5**	**50**	0.93	0.99	0.98	0.98	0.98
150	**50**	**100**	0.96	0.92	0.94	0.94	0.93	**250**	**200**	**100**	0.90	0.98	0.94	0.94	0.94		**1000**	**5**	**100**	0.95	0.99	0.98	0.98	0.98
150	**100**	**5**	0.99	0.97	0.95	0.95	0.95	**500**	**5**	**5**	**0.98**	**0.98**	**0.98**	**0.98**	**0.98**	****(0.98)**	**1000**	**50**	**5**	0.98	0.95	0.94	0.94	0.93
150	**100**	**10**	0.97	0.98	0.95	0.95	0.95	**500**	**5**	**10**	0.98	0.98	0.97	0.97	0.96		**1000**	**50**	**10**	0.95	0.95	0.94	0.94	0.93
150	**100**	**50**	0.94	0.98	0.95	0.95	0.95	**500**	**5**	**50**	0.97	0.96	0.96	0.96	0.94		**1000**	**50**	**50**	0.97	0.95	0.94	0.94	0.93
150	**100**	**100**	0.94	0.98	0.96	0.96	0.96	**500**	**5**	**100**	0.98	0.94	0.96	0.96	0.94		**1000**	**50**	**100**	0.94	0.94	0.94	0.94	0.93
250	**5**	**5**	0.96	0.98	0.99	0.99	0.99	**500**	**50**	**5**	0.95	0.97	0.96	0.94	0.94		**1000**	**100**	**5**	0.92	0.94	0.91	0.91	0.91
250	**5**	**10**	0.97	0.94	0.94	0.96	0.93	**500**	**50**	**10**	0.91	0.97	0.95	0.94	0.94		**1000**	**100**	**10**	0.94	0.95	0.91	0.92	0.92
250	**5**	**50**	0.98	0.94	0.94	0.96	0.93	**500**	**50**	**50**	0.93	0.97	0.93	0.94	0.92		**1000**	**100**	**50**	0.94	0.97	0.92	0.92	0.94
250	**5**	**100**	0.94	0.94	0.94	0.96	0.93	**500**	**50**	**100**	0.94	0.97	0.96	0.97	0.95		**1,000**	**100**	**100**	0.96	0.95	0.93	0.94	0.94
250	**50**	**5**	0.99	0.97	0.94	0.95	0.94	**500**	**100**	**5**	0.97	0.98	0.95	0.95	0.95		**1000**	**200**	**5**	0.93	0.97	0.93	0.93	0.93
250	**50**	**10**	0.96	0.97	0.94	0.95	0.94	**500**	**100**	**10**	0.94	0.98	0.95	0.95	0.95		**1000**	**200**	**10**	0.92	0.98	0.93	0.93	0.93
250	**50**	**50**	0.97	0.95	0.94	0.94	0.93	**500**	**100**	**50**	0.93	0.95	0.94	0.93	0.94		**1000**	**200**	**50**	0.94	0.98	0.94	0.94	0.95
250	**50**	**100**	0.96	0.93	0.93	0.94	0.93	**500**	**100**	**100**	0.90	0.98	0.95	0.95	0.95		**1000**	**200**	**100**	0.95	0.95	0.93	0.93	0.93

Table E.6: Inner Loop Performances for p_{rc}, p_{lc} (PCM+NSGA-II, Rareness Level=10%)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	**0.98**	**0.98**	**0.98**	**0.98**	**0.98**	****(0.98)**
0.6-0.4	0.95	0.99	0.98	0.98	0.98	**(0.97)
0.8-0.2	0.95	0.97	0.98	0.98	0.98	
1.0-0.0	0.95	0.95	0.97	0.98	0.97	
0.4-0.6	0.95	0.95	0.98	0.98	0.98	
0.2-0.8	0.92	0.96	0.97	0.97	0.96	
0.0-1.0	0.94	0.96	0.96	0.96	0.97	

Table E.7: Inner Loop Performances for p_m (PCM+NSGA-II, Rareness Level=10%)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.98**	**0.98**	**0.98**	**0.98**	**0.98**	***(0.98)**
0.05	0.90	0.95	0.97	0.96	0.95	
0.10	0.94	0.98	0.97	0.97	0.97	
0.50	0.94	0.80	0.78	0.80	0.78	

E.2 Hyper-parameter Optimization for PCM+RECGA

E.2.1 Rareness Level = 1%

Table E.8 indicates the Fscore performances of inner cross validation for different values of *PopulationSize* and *minFinalSetSize*.

The hyper-parameter values whose inner cross validation performances are among

the top two for all the given test folds are marked with ** in the last column. Among the set of hyper-parameters that has highest average performances, *PopulationSize*, and *NumberOfGenerations* are set to 150 and 50, respectively.

Then we repeat the same analysis for p_{rc} and p_{lc}. The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column of Table E.9 and we set p_{rc} and p_{lc} as 0.4 and 0.6, respectively.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top two for all the given test folds are marked with ** in the last column of Table E.10. Due to its higher average performance, we set p_m = 0.01.

Table E.8: Inner Loop Performances for *PopulationSize* and *minFinalSetSize* (PCM+RECGA, Rareness Level=1%)

PopulationSize	**minFinalSetSize**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	0.98	0.98	0.98	0.98	0.98	**(0.98)
150	**50**	**0.98**	**0.98**	**0.98**	**0.98**	**0.98**	****(0.98)**
150	**100**	0.98	0.98	0.98	0.98	0.98	**(0.98)
250	**5**	0.75	0.99	0.98	0.98	0.98	**(0.94)
250	**50**	0.75	0.99	0.98	0.98	0.98	**(0.94)
250	**100**	0.75	0.98	0.98	0.98	0.98	**(0.94)
250	**200**	0.75	0.98	0.98	0.98	0.98	**(0.94)
500	**5**	0.75	0.99	0.98	0.99	0.99	**(0.94)
500	**50**	0.75	0.99	0.98	0.99	0.99	**(0.94)
500	**100**	0.75	0.99	0.98	0.99	0.99	**(0.94)
500	**200**	0.75	0.99	0.98	0.99	0.99	**(0.94)
1000	**5**	0.75	0.99	0.98	0.99	0.98	**(0.94)
1000	**50**	0.50	0.99	0.98	0.99	0.98	
1000	**100**	0.50	0.99	0.98	0.99	0.98	
1000	**200**	0.50	0.99	0.98	0.99	0.99	

Table E.9: Inner Loop Performances for p_{rc}, p_{lc} (PCM+RECGA, Rareness Level=1%)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	0.98	0.98	0.98	0.98	0.98	*(0.98)
0.6-0.4	0.74	0.98	0.98	0.98	0.98	
0.8-0.2	0.74	0.98	0.98	0.98	0.98	
1.0-0.0	0.98	0.98	0.98	0.98	0.98	*(0.98)
0.4-0.6	**0.98**	**0.98**	**0.98**	**0.98**	**0.98**	***(0.98)**
0.2-0.8	0.98	0.98	0.98	0.98	0.98	*(0.98)
0.0-1.0	0.98	0.98	0.98	0.98	0.98	*(0.98)

Table E.10: Inner Loop Performances for p_m (PCM+RECGA, Rareness Level=1%)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.98**	**0.98**	**0.98**	**0.98**	**0.98**	****(0.98)**
0.05	0.74	0.98	0.98	0.98	0.98	**(0.93)
0.10	0.74	0.99	0.98	0.99	0.99	**(0.94)
0.50	0.74	0.98	0.98	0.98	0.98	**(0.93)

E.2.2 Rareness Level = 10%

Table E.11 indicates Fscore performances of inner cross validation for different values of *PopulationSize* and *minFinalSetSize*.

The hyper-parameter values whose inner cross validation performances are among the top one for all the given test folds are marked with * in the last column. *PopulationSize*, and *NumberOfGenerations* are set to 250 and 200, respectively.

Then we repeat the same analysis for p_{rc} and p_{lc}. The hyper-parameter values whose inner cross validation performances are among the top one for all the given test folds are marked with * in the last column of Table E.12 and we set p_{rc} and p_{lc} as 0.8 and 0.2, respectively.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top one for all the given test folds are marked with * in the last column of Table E.13 and we set p_m = 0.01.

Table E.11: Inner Loop Performances for *PopulationSize* and *minFinalSetSize* (PCM+RECGA, Rareness Level=10%)

PopulationSize	**minFinalSetSize**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	0.94	0.94	0.94	0.94	0.93	*(0.94)
150	**50**	0.94	0.94	0.94	0.94	0.93	*(0.94)
150	**100**	0.94	0.94	0.94	0.94	0.93	*(0.94)
250	**5**	0.94	0.94	0.94	0.94	0.93	*(0.94)
250	**50**	0.94	0.94	0.94	0.94	0.93	*(0.94)
250	**100**	0.94	0.94	0.94	0.94	0.93	*(0.94)
250	**200**	**0.94**	**0.94**	**0.94**	**0.94**	**0.93**	***(0.94)**
500	**5**	0.92	0.94	0.94	0.94	0.93	
500	**50**	0.92	0.94	0.94	0.94	0.93	
500	**100**	0.92	0.94	0.94	0.94	0.93	
500	**200**	0.92	0.94	0.94	0.94	0.93	
1000	**5**	0.94	0.94	0.94	0.94	0.93	*(0.94)
1000	**50**	0.94	0.94	0.94	0.94	0.93	*(0.94)
1000	**100**	0.94	0.94	0.94	0.94	0.93	*(0.94)
1000	**200**	0.94	0.94	0.94	0.94	0.93	*(0.94)

Table E.12: Inner Loop Performances for p_{rc}, p_{lc} (PCM+RECGA, Rareness Level=10%)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	0.94	0.94	0.94	0.94	0.93	
0.6-0.4	0.94	0.94	0.94	0.94	0.93	
0.8-0.2	**0.94**	**0.95**	**0.95**	**0.95**	**0.95**	***(0.95)**
1.0-0.0	0.94	0.95	0.95	0.95	0.94	
0.4-0.6	0.94	0.94	0.94	0.94	0.93	
0.2-0.8	0.94	0.94	0.94	0.94	0.93	
0.0-1.0	0.94	0.95	0.95	0.95	0.95	*(0.95)

Table E.13: Inner Loop Performances for p_m (PCM+RECGA, Rareness Level=10%)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.94**	**0.95**	**0.95**	**0.95**	**0.95**	***(0.95)**
0.05	0.94	0.89	0.90	0.90	0.91	
0.10	0.94	0.94	0.93	0.94	0.93	
0.50	0.94	0.84	0.87	0.88	0.87	

F Generalization Performances of the Models for the Wisconsin Breast Cancer Original Dataset: 5-fold Cross Validation

To see the generalization performance of the models with the selected hyper-parameter values, we test the models in outer loop of the nested cross validation. Thus, we test the model performances with 5-fold cross validation. We repeat the experiments for the rareness levels ranging between 1% to 35%.

Table F.1 shows the dataset configurations used in the experiments which are conducted with 5-fold CV.

Table F.1: Experimental Settings for the WBCO Dataset (5-fold CV)

	Malign	**Benign**	**Rareness level**		**Malign**	**Benign**	**Rareness level**
$\mathcal{S}$	2	176		$\mathcal{S}$	20	176	
$\mathcal{V}$	2	176		$\mathcal{V}$	20	176	
$\widetilde{\mathcal{S}}$	1	88	**1%**	$\widetilde{\mathcal{S}}$	10	88	**10%**
Total	5	440		Total	50	440	
$\mathcal{S}$	6	176		$\mathcal{S}$	32	176	
$\mathcal{V}$	6	176		$\mathcal{V}$	32	176	
$\widetilde{\mathcal{S}}$	3	88	**3%**	$\widetilde{\mathcal{S}}$	16	88	**15%**
Total	15	440		Total	80	440	
$\mathcal{S}$	10	176		$\mathcal{S}$	60	176	
$\mathcal{V}$	10	176		$\mathcal{V}$	60	176	
$\widetilde{\mathcal{S}}$	5	88	**5%**	$\widetilde{\mathcal{S}}$	30	88	**25%**
Total	25	440		Total	150	440	
$\mathcal{S}$	14	176		$\mathcal{S}$	94	176	
$\mathcal{V}$	14	176		$\mathcal{V}$	94	176	
$\widetilde{\mathcal{S}}$	7	88	**7%**	$\widetilde{\mathcal{S}}$	47	88	**35%**
Total	35	440		Total	235	440	
L			$\{0,\ldots 176\}$				

Tables F.2 and F.3 summarize the average performance results and standard deviations of performance indicators for PCM+NSGA-II and PCM+RECGA. The performances

are given for both sets of hyper-parameters, that are the hyper-parameters determined with rareness level 1% and 10%.

We also report the performances of the competitor models under the experimental setting with 5-fold cross validation, for the sake of completeness. Their average training and test performances are given in Table F.4 and standard deviations of performance indicators for training and test are given in Table F.5.

Table F.2: Performances of PCM+NSGA-II (5-fold CV, WBCO Dataset)

PCM+NSGA-II	1%		3%		5%		7%		10%		15%		25%		35%	
	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
AVERAGE PERFORMANCE RESULTS																
$\mathcal{S}$																
Sensitivity	1.00	0.90	0.80	0.50	0.92	0.72	0.66	0.84	0.73	0.83	0.97	0.95	0.97	0.82	0.95	0.93
Specificity	0.94	0.99	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.98	0.98	0.98	0.98	0.98	0.97	0.98
Accuracy	0.94	0.99	0.97	0.97	0.98	0.97	0.97	0.98	0.96	0.97	0.98	0.98	0.97	0.94	0.97	0.96
Fscore	0.97	0.93	0.86	0.65	0.95	0.83	0.76	0.90	0.84	0.90	0.97	0.96	0.97	0.88	0.96	0.95
Fmeasure	0.29	0.59	0.64	0.50	0.81	0.73	0.72	0.84	0.80	0.84	0.93	0.92	0.95	0.86	0.95	0.94
$\mathcal{V}$																
Sensitivity	1.00	0.50	0.77	0.83	0.84	0.88	0.84	0.93	0.84	0.91	0.91	0.97	0.93	0.84	0.94	0.92
Specificity	0.96	1.00	0.98	1.00	0.98	0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.98	0.98	0.98	0.99
Accuracy	0.96	0.99	0.97	0.99	0.98	0.99	0.98	0.99	0.98	0.98	0.97	0.99	0.97	0.95	0.96	0.96
Fscore	0.98	0.67	0.86	0.91	0.90	0.93	0.91	0.96	0.91	0.95	0.95	0.98	0.95	0.90	0.96	0.95
Fmeasure	0.35	0.55	0.62	0.89	0.79	0.89	0.84	0.92	0.87	0.92	0.91	0.96	0.93	0.88	0.95	0.94
$\widetilde{\mathcal{S}}$																
Sensitivity	1.00	0.60	0.87	0.67	0.84	0.80	0.80	0.86	0.78	0.84	0.85	0.90	0.93	0.81	0.95	0.90
Specificity	0.95	0.99	0.96	0.98	0.98	0.98	0.98	0.98	0.98	0.97	0.98	0.98	0.97	0.98	0.97	0.97
Accuracy	0.95	0.98	0.96	0.97	0.97	0.97	0.96	0.97	0.96	0.96	0.96	0.97	0.96	0.94	0.96	0.94
Fscore	0.97	0.60	0.90	0.75	0.89	0.87	0.87	0.91	0.85	0.90	0.91	0.94	0.95	0.88	0.96	0.93
Fmeasure	0.31	0.37	0.60	0.58	0.76	0.75	0.76	0.81	0.80	0.81	0.88	0.90	0.92	0.86	0.95	0.92
STANDARD DEVIATIONS OF PERFORMANCE INDICATORS																
$\mathcal{S}$																
Sensitivity	0.00	0.20	0.24	0.15	0.07	0.12	0.24	0.15	0.09	0.09	0.02	0.04	0.02	0.15	0.02	0.03
Specificity	0.03	0.01	0.01	0.01	0.01	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.01	0.01	0.01	0.02
Accuracy	0.03	0.01	0.00	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.00	0.01	0.01	0.04	0.01	0.01
Fscore	0.01	0.13	0.16	0.13	0.04	0.08	0.18	0.09	0.06	0.06	0.01	0.02	0.01	0.10	0.01	0.01
Fmeasure	0.09	0.10	0.08	0.13	0.08	0.08	0.16	0.09	0.05	0.08	0.01	0.02	0.01	0.10	0.01	0.01
$\mathcal{V}$																
Sensitivity	0.00	0.00	0.08	0.00	0.08	0.04	0.03	0.05	0.10	0.06	0.03	0.00	0.01	0.13	0.01	0.05
Specificity	0.01	0.00	0.01	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Accuracy	0.01	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.01	0.00	0.01	0.00	0.01	0.03	0.01	0.02
Fscore	0.01	0.00	0.05	0.00	0.05	0.02	0.02	0.02	0.06	0.03	0.02	0.00	0.01	0.08	0.01	0.03
Fmeasure	0.07	0.10	0.03	0.03	0.05	0.02	0.03	0.04	0.06	0.03	0.02	0.01	0.01	0.08	0.01	0.02
$\widetilde{\mathcal{S}}$																
Sensitivity	0.00	0.49	0.16	0.30	0.20	0.18	0.15	0.09	0.19	0.08	0.06	0.03	0.05	0.13	0.02	0.06
Specificity	0.02	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.02	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.03	0.01	0.01
Fscore	0.01	0.49	0.10	0.22	0.12	0.11	0.09	0.05	0.14	0.05	0.04	0.02	0.02	0.08	0.01	0.03
Fmeasure	0.07	0.31	0.13	0.14	0.05	0.08	0.08	0.08	0.12	0.04	0.04	0.02	0.02	0.07	0.01	0.02

Table F.3: Performances of PCM+RECGA (5-fold CV, WBCO Dataset)

PCM+RECGA	1%		3%		5%		7%		10%		15%		25%		35%	
	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
AVERAGE PERFORMANCE RESULTS																
$\mathcal{S}$																
Sensitivity	1.00	1.00	0.73	0.70	0.86	0.86	0.87	0.89	0.67	0.70	0.99	0.99	0.99	0.99	0.98	0.98
Specificity	0.96	0.96	1.00	1.00	0.97	0.97	0.98	0.96	0.98	0.97	0.90	0.90	0.94	0.94	0.95	0.95
Accuracy	0.96	0.96	0.99	0.99	0.97	0.96	0.97	0.96	0.95	0.95	0.91	0.91	0.95	0.95	0.96	0.96
Fscore	0.98	0.98	0.84	0.82	0.90	0.90	0.92	0.92	0.79	0.81	0.94	0.94	0.96	0.96	0.97	0.97
Fmeasure	0.37	0.37	0.78	0.76	0.73	0.73	0.80	0.76	0.72	0.73	0.78	0.78	0.92	0.92	0.95	0.95
$\mathcal{V}$																
Sensitivity	1.00	1.00	0.93	0.93	0.96	0.98	0.89	0.90	0.77	0.81	0.95	0.95	0.93	0.93	0.94	0.94
Specificity	0.96	0.96	0.98	0.98	0.96	0.96	0.97	0.97	0.98	0.98	0.91	0.91	0.93	0.92	0.96	0.96
Accuracy	0.96	0.96	0.98	0.98	0.96	0.96	0.97	0.97	0.96	0.96	0.91	0.91	0.93	0.93	0.95	0.95
Fscore	0.98	0.98	0.95	0.95	0.96	0.97	0.92	0.93	0.85	0.88	0.93	0.93	0.93	0.93	0.95	0.95
Fmeasure	0.36	0.36	0.74	0.75	0.74	0.74	0.79	0.80	0.78	0.81	0.78	0.78	0.87	0.87	0.93	0.93
$\widetilde{\mathcal{S}}$																
Sensitivity	1.00	1.00	0.80	0.60	0.84	0.88	0.86	0.89	0.78	0.80	0.96	0.96	0.95	0.95	0.94	0.94
Specificity	0.95	0.95	0.98	0.98	0.95	0.95	0.97	0.97	0.98	0.97	0.90	0.90	0.91	0.91	0.97	0.96
Accuracy	0.95	0.95	0.97	0.97	0.95	0.95	0.96	0.96	0.96	0.95	0.91	0.91	0.92	0.92	0.96	0.96
Fscore	0.98	0.98	0.85	0.69	0.88	0.91	0.90	0.92	0.86	0.87	0.93	0.93	0.93	0.93	0.95	0.95
Fmeasure	0.33	0.33	0.64	0.51	0.64	0.67	0.74	0.77	0.79	0.77	0.78	0.78	0.87	0.86	0.94	0.94
STANDARD DEVIATIONS OF PERFORMANCE INDICATORS																
$\mathcal{S}$																
Sensitivity	0.00	0.00	0.08	0.12	0.17	0.17	0.07	0.07	0.06	0.00	0.03	0.03	0.01	0.01	0.00	0.00
Specificity	0.01	0.01	0.00	0.00	0.02	0.02	0.01	0.02	0.01	0.00	0.06	0.06	0.03	0.04	0.01	0.01
Accuracy	0.01	0.01	0.00	0.01	0.02	0.02	0.00	0.02	0.00	0.00	0.05	0.05	0.02	0.02	0.01	0.01
Fscore	0.00	0.00	0.05	0.09	0.11	0.10	0.04	0.04	0.04	0.00	0.03	0.03	0.02	0.02	0.00	0.01
Fmeasure	0.05	0.05	0.07	0.11	0.15	0.14	0.02	0.08	0.02	0.02	0.10	0.10	0.04	0.04	0.01	0.01
$\mathcal{V}$																
Sensitivity	0.00	0.00	0.08	0.08	0.05	0.04	0.09	0.10	0.18	0.12	0.04	0.04	0.02	0.02	0.03	0.04
Specificity	0.01	0.01	0.00	0.00	0.02	0.02	0.01	0.01	0.01	0.01	0.05	0.05	0.05	0.04	0.01	0.01
Accuracy	0.01	0.01	0.00	0.00	0.02	0.02	0.00	0.00	0.01	0.01	0.04	0.04	0.03	0.03	0.01	0.01
Fscore	0.01	0.01	0.04	0.04	0.02	0.02	0.05	0.05	0.12	0.07	0.02	0.02	0.02	0.02	0.01	0.01
Fmeasure	0.07	0.07	0.02	0.02	0.09	0.09	0.01	0.03	0.09	0.04	0.08	0.08	0.05	0.04	0.01	0.01
$\widetilde{\mathcal{S}}$																
Sensitivity	0.00	0.00	0.27	0.33	0.20	0.16	0.16	0.17	0.17	0.14	0.05	0.05	0.03	0.03	0.05	0.05
Specificity	0.01	0.01	0.00	0.00	0.03	0.03	0.00	0.00	0.01	0.01	0.05	0.05	0.06	0.05	0.01	0.01
Accuracy	0.01	0.01	0.01	0.01	0.02	0.02	0.01	0.01	0.01	0.02	0.05	0.05	0.05	0.04	0.02	0.01
Fscore	0.01	0.01	0.19	0.24	0.11	0.09	0.10	0.10	0.11	0.09	0.04	0.04	0.04	0.04	0.02	0.02
Fmeasure	0.05	0.05	0.14	0.19	0.07	0.10	0.09	0.11	0.08	0.10	0.09	0.09	0.07	0.07	0.02	0.02

Table F.4: Average Performance Results of Competitor Models (5-fold CV, WBCO Dataset)

	LR	pen-LR	SVM	ANN	DT	RF	LR	pen-LR	SVM	ANN	DT	RF
			$\mathcal{S} \cup \mathcal{V}$						$\widetilde{\mathcal{S}}$			
						1%						
Sensitivity	1.00	0.80	0.35	1.00	0.35	1.00	0.60	0.60	0.00	0.80	0.00	0.00
Specificity	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	1.00	0.99	0.99	0.99
Accuracy	1.00	1.00	0.99	1.00	0.99	1.00	0.98	0.99	0.98	0.99	0.98	0.98
Fscore	1.00	0.89	0.37	1.00	0.41	1.00	0.60	0.60	0.00	0.79	0.00	0.00
Fmeasure	1.00	0.84	0.37	0.96	0.37	1.00	0.50	0.50	0.00	0.58	0.00	0.00
						3%						
Sensitivity	0.92	0.92	0.73	0.97	0.92	1.00	0.60	0.67	0.47	0.67	0.53	0.67
Specificity	1.00	1.00	1.00	1.00	1.00	1.00	0.98	0.99	0.99	0.98	1.00	1.00
Accuracy	0.99	0.99	0.99	1.00	1.00	1.00	0.97	0.98	0.97	0.97	0.98	0.98
Fscore	0.95	0.95	0.76	0.98	0.95	1.00	0.71	0.77	0.58	0.78	0.56	0.72
Fmeasure	0.91	0.91	0.73	0.97	0.94	1.00	0.56	0.64	0.46	0.65	0.50	0.67
						5%						
Sensitivity	0.95	0.94	0.93	0.98	0.92	1.00	0.84	0.76	0.80	0.88	0.72	0.72
Specificity	0.99	0.99	0.99	1.00	0.99	1.00	0.99	0.99	0.99	0.99	0.98	0.99
Accuracy	0.99	0.99	0.99	1.00	0.99	1.00	0.98	0.98	0.98	0.98	0.97	0.97
Fscore	0.97	0.97	0.96	0.99	0.95	1.00	0.91	0.84	0.87	0.93	0.81	0.81
Fmeasure	0.93	0.92	0.90	0.96	0.89	1.00	0.84	0.78	0.78	0.83	0.69	0.72
						7%						
Sensitivity	0.92	0.93	0.88	0.95	0.96	1.00	0.86	0.83	0.91	0.83	0.80	0.77
Specificity	0.99	0.99	0.99	0.99	0.99	1.00	0.98	0.99	0.98	0.99	0.98	0.98
Accuracy	0.99	0.99	0.98	0.99	0.99	1.00	0.97	0.97	0.98	0.97	0.97	0.96
Fscore	0.95	0.96	0.93	0.97	0.97	1.00	0.91	0.90	0.95	0.90	0.87	0.81
Fmeasure	0.90	0.92	0.88	0.94	0.91	1.00	0.83	0.83	0.87	0.83	0.78	0.71
						10%						
Sensitivity	0.95	0.95	0.94	0.98	0.96	1.00	0.86	0.84	0.92	0.86	0.74	0.82
Specificity	0.99	0.99	0.99	0.99	0.99	1.00	0.98	0.98	0.99	0.98	0.99	0.98
Accuracy	0.98	0.99	0.98	0.99	0.99	1.00	0.97	0.97	0.98	0.97	0.96	0.97
Fscore	0.97	0.97	0.96	0.99	0.97	1.00	0.91	0.90	0.95	0.91	0.83	0.89
Fmeasure	0.93	0.93	0.91	0.96	0.94	1.00	0.86	0.84	0.90	0.85	0.79	0.83
						15%						
Sensitivity	0.93	0.92	0.93	0.99	0.96	1.00	0.89	0.86	0.94	0.86	0.84	0.93
Specificity	0.98	0.99	0.98	0.99	0.99	1.00	0.99	0.99	0.98	0.98	0.98	0.98
Accuracy	0.98	0.98	0.98	0.99	0.98	1.00	0.97	0.97	0.98	0.96	0.96	0.97
Fscore	0.96	0.95	0.95	0.99	0.97	1.00	0.93	0.92	0.96	0.92	0.90	0.95
Fmeasure	0.92	0.92	0.92	0.96	0.95	1.00	0.91	0.89	0.92	0.87	0.85	0.91
						25%						
Sensitivity	0.95	0.94	0.95	1.00	0.95	1.00	0.92	0.91	0.93	0.95	0.91	0.92
Specificity	0.98	0.98	0.98	0.98	0.99	1.00	0.98	0.98	0.97	0.95	0.96	0.97
Accuracy	0.97	0.97	0.97	0.98	0.98	1.00	0.96	0.96	0.96	0.95	0.95	0.96
Fscore	0.96	0.96	0.97	0.99	0.97	1.00	0.95	0.94	0.95	0.95	0.93	0.94
Fmeasure	0.95	0.95	0.95	0.97	0.96	1.00	0.93	0.92	0.92	0.91	0.90	0.92
						35%						
Sensitivity	0.96	0.96	0.97	1.00	0.98	1.00	0.95	0.94	0.96	0.96	0.94	0.97
Specificity	0.98	0.98	0.97	0.98	0.98	1.00	0.97	0.97	0.97	0.96	0.96	0.97
Accuracy	0.97	0.97	0.97	0.99	0.98	1.00	0.96	0.96	0.97	0.96	0.95	0.97
Fscore	0.97	0.97	0.97	0.99	0.98	1.00	0.96	0.96	0.96	0.96	0.95	0.97
Fmeasure	0.96	0.96	0.96	0.98	0.97	1.00	0.95	0.94	0.95	0.94	0.93	0.96

Table F.5: Standard Deviations of Performance Indicators of Competitor Models (5-fold CV, WBCO Dataset)

	LR	pen-LR	SVM	ANN	DT	RF	LR	pen-LR	SVM	ANN	DT	RF
			$\mathcal{S} \cup \mathcal{V}$						$\widetilde{\mathcal{S}}$			
						1%						
Sensitivity	0.00	0.10	0.44	0.00	0.37	0.00	0.49	0.49	0.00	0.40	0.00	0.00
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.00	0.06	0.46	0.00	0.39	0.00	0.49	0.49	0.00	0.40	0.00	0.00
Fmeasure	0.00	0.05	0.46	0.05	0.33	0.00	0.45	0.45	0.00	0.38	0.00	0.00
						3%						
Sensitivity	0.05	0.05	0.37	0.04	0.09	0.00	0.25	0.21	0.27	0.21	0.45	0.37
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01	0.01	0.02	0.01	0.01
Accuracy	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.01	0.01	0.02	0.01	0.01
Fscore	0.03	0.03	0.38	0.02	0.05	0.00	0.19	0.16	0.31	0.16	0.46	0.37
Fmeasure	0.06	0.04	0.36	0.03	0.04	0.00	0.19	0.12	0.24	0.21	0.43	0.37
						5%						
Sensitivity	0.03	0.02	0.04	0.02	0.07	0.00	0.08	0.20	0.18	0.10	0.20	0.24
Specificity	0.00	0.00	0.01	0.00	0.01	0.00	0.01	0.01	0.01	0.01	0.02	0.01
Accuracy	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.02
Fscore	0.02	0.01	0.02	0.01	0.04	0.00	0.04	0.14	0.11	0.05	0.14	0.16
Fmeasure	0.04	0.02	0.06	0.02	0.03	0.00	0.07	0.12	0.13	0.09	0.10	0.19
						7%						
Sensitivity	0.01	0.02	0.02	0.04	0.04	0.00	0.09	0.06	0.07	0.11	0.17	0.33
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.00
Accuracy	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.02
Fscore	0.01	0.01	0.01	0.02	0.02	0.00	0.05	0.03	0.04	0.06	0.10	0.29
Fmeasure	0.01	0.01	0.02	0.01	0.01	0.00	0.06	0.03	0.06	0.08	0.08	0.25
						10%						
Sensitivity	0.01	0.01	0.01	0.02	0.04	0.00	0.10	0.14	0.04	0.12	0.16	0.13
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.01	0.01	0.01	0.02
Fscore	0.01	0.01	0.01	0.01	0.02	0.00	0.06	0.09	0.02	0.07	0.10	0.08
Fmeasure	0.01	0.01	0.02	0.01	0.02	0.00	0.09	0.11	0.04	0.07	0.08	0.10
						15%						
Sensitivity	0.01	0.01	0.01	0.03	0.02	0.00	0.03	0.03	0.04	0.06	0.05	0.03
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01
Accuracy	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.02	0.02	0.03	0.03	0.01
Fmeasure	0.01	0.01	0.01	0.01	0.02	0.00	0.03	0.04	0.04	0.04	0.04	0.03
						25%						
Sensitivity	0.01	0.01	0.01	0.00	0.02	0.00	0.05	0.04	0.04	0.03	0.02	0.03
Specificity	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.01	0.02	0.01	0.01
Accuracy	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.01	0.01	0.00	0.00	0.01	0.00	0.02	0.02	0.02	0.01	0.01	0.01
Fmeasure	0.01	0.01	0.00	0.01	0.01	0.00	0.02	0.02	0.02	0.02	0.01	0.01
						35%						
Sensitivity	0.01	0.01	0.01	0.00	0.01	0.00	0.03	0.04	0.01	0.03	0.03	0.02
Specificity	0.00	0.00	0.00	0.01	0.00	0.00	0.02	0.02	0.01	0.02	0.02	0.01
Accuracy	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.01	0.01
Fscore	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.01	0.01	0.01	0.01
Fmeasure	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.02	0.01	0.02	0.02	0.01

Note that the given performances of PCM+NSGA-II and PCM+RECGA are similar to those of the results of experimental analyses performed with randomly generated 100 instances, except for configurations where the rareness level is extremely low. For such configurations, since the samples consist of very few positive observations and the experiments are repeated only five times in 5-fold cross validation, the results are susceptible to these factors. For such configurations, repeating the experiments as much as possible reflects the model performances' better.

G Hyper-parameter Optimization for the Wisconsin Breast Cancer Diagnostic Dataset

Table G.1 summarizes the content of a fold. Note that, the positive and negative observations in a fold are arranged to satisfy predetermined rareness levels. Columns "Malign" and "Benign" of Table G.1 indicate the number of positive and negative observations in a fold and "Rareness level" indicates the ratio of positive observations to all observations.

Table G.1: Content of a Fold for Different Rareness Levels - WBCD Dataset

		Malign	Benign
Rareness level	1%	1	70
	3%	2	70
	5%	4	70
	7%	5	70
	10%	8	70
	15%	12	70
	25%	23	70
	37%	41	70

G.1 Hyper-parameter Optimization for PCM+NSGA-II

G.1.1 Rareness Level = 1%

Table G.2 indicates the Fscore performances of inner cross validation for different values of *PopulationSize*, *GenerationSize* and *NumberOfGenerations*.

The hyper-parameter values whose inner cross validation performances are among the top three for all the given test folds are marked with *** in the last column. Due to the higher average value, the *PopulationSize*, *GenerationSize* and *NumberOfGenerations* are set to 150, 50 and 5, respectively. Once these values are set, we repeat the same analysis for p_{rc} and p_{lc}.

The hyper-parameter values whose inner cross validation performances are among the top three for all the given test folds are marked with *** in the last column of Table G.3. Since it provides a higher average performance, we set $p_{rc} = 0.5$ and $p_{lc} = 0.5$.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top one for all the given test folds are marked with * in the last column of Table G.4. Thus we set p_m=0.10.

Table G.2: Inner Loop Performances for *PopulationSize*, *GenerationSize* and *NumberOfGenerations* (PCM+NSGA-II, Rareness Level=1%)

PopulationSize	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5		**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5
150	**5**	**5**	0.99	0.99	0.74	0.74	0.75		**250**	**100**	**5**	0.94	0.96	0.74	0.74	0.74	**500**	**200**	**5**	0.90	0.90	0.93	0.93	0.93
150	**5**	**10**	0.75	0.99	0.99	0.99	1.00		**250**	**100**	**10**	0.95	0.97	0.74	0.73	0.74	**500**	**200**	**10**	0.89	0.91	0.70	0.70	0.70
150	**5**	**50**	0.75	0.75	0.50	0.50	0.75		**250**	**100**	**50**	0.94	0.96	0.73	0.73	0.74	**500**	**200**	**50**	0.93	0.71	0.47	0.47	0.48
150	**5**	**100**	0.75	0.75	0.50	0.50	0.75		**250**	**100**	**100**	0.94	0.97	0.74	0.73	0.74	**500**	**200**	**100**	0.73	0.95	0.71	0.70	0.71
150	**50**	**5**	**0.97**	**0.98**	**0.98**	**0.99**	**0.99**	***(0.982)**	**250**	**200**	**5**	0.93	0.94	0.72	0.72	0.73	**1000**	**5**	**5**	0.49	0.50	0.25	0.00	0.25
150	**50**	**10**	0.98	0.97	0.98	0.98	0.99	***(0.980)	**250**	**200**	**10**	0.89	0.92	0.71	0.71	0.71	**1000**	**5**	**10**	0.50	0.50	0.50	0.50	0.75
150	**50**	**50**	0.98	0.98	0.74	0.74	0.74		**250**	**200**	**50**	0.88	0.94	0.73	0.73	0.73	**1000**	**5**	**50**	0.50	0.50	0.50	0.50	0.75
150	**50**	**100**	0.74	0.98	0.49	0.49	0.50		**250**	**200**	**100**	0.87	0.95	0.73	0.73	0.73	**1000**	**5**	**100**	0.50	0.50	0.50	0.50	0.75
150	**100**	**5**	0.97	0.96	0.98	0.98	0.98		**500**	**5**	**5**	0.74	0.99	0.74	0.74	0.75	**1000**	**50**	**5**	0.98	0.98	0.74	0.74	0.74
150	**100**	**10**	0.94	0.95	0.97	0.98	0.98		**500**	**5**	**10**	0.74	0.99	0.74	0.74	0.75	**1000**	**50**	**10**	0.97	0.98	0.74	0.74	0.74
150	**100**	**50**	0.91	0.96	0.74	0.74	0.74		**500**	**5**	**50**	0.50	0.50	0.50	0.50	0.75	**1000**	**50**	**50**	0.98	0.74	0.74	0.74	0.74
150	**100**	**100**	0.93	0.96	0.74	0.73	0.74		**500**	**5**	**100**	0.50	0.50	0.50	0.50	0.75	**1000**	**50**	**100**	0.97	0.99	0.74	0.74	0.74
250	**5**	**5**	0.75	0.99	0.99	0.99	1.00		**500**	**50**	**5**	0.98	0.99	0.74	0.74	0.75	**1000**	**100**	**5**	0.93	0.96	0.74	0.74	0.74
250	**5**	**10**	0.75	0.99	0.99	0.99	0.99		**500**	**50**	**10**	0.99	0.98	0.49	0.49	0.50	**1000**	**100**	**10**	0.92	0.96	0.73	0.73	0.74
250	**5**	**50**	0.75	0.75	0.50	0.50	0.75		**500**	**50**	**50**	0.96	0.99	0.74	0.74	0.74	**1000**	**100**	**50**	0.92	0.74	0.73	0.73	0.74
250	**5**	**100**	0.75	0.75	0.50	0.50	0.75		**500**	**50**	**100**	0.48	0.75	1.00	1.00	1.00	**1,000**	**100**	**100**	0.72	0.73	0.73	0.73	0.74
250	**50**	**5**	0.97	0.99	0.49	0.49	0.50		**500**	**100**	**5**	0.94	0.96	0.74	0.74	0.74	**1000**	**200**	**5**	0.88	0.92	0.72	0.70	0.72
250	**50**	**10**	0.98	0.98	0.74	0.74	0.75		**500**	**100**	**10**	0.95	0.98	0.74	0.74	0.74	**1000**	**200**	**10**	0.86	0.92	0.71	0.71	0.71
250	**50**	**50**	0.98	0.98	0.74	0.74	0.75		**500**	**100**	**50**	0.97	0.98	0.49	0.49	0.50	**1000**	**200**	**50**	0.65	0.93	0.71	0.71	0.72
250	**50**	**100**	0.98	0.99	0.49	0.49	0.50		**500**	**100**	**100**	0.99	0.98	0.49	0.49	0.50	**1000**	**200**	**100**	0.66	0.72	0.72	0.72	0.73

Table G.3: Inner Loop Performances for p_{rc}, p_{lc} (PCM+NSGA-II, Rareness Level=1%)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	**0.97**	**0.98**	**0.98**	**0.99**	**0.99**	***(0.98)**
0.6-0.4	0.98	0.99	0.74	0.74	0.75	***(0.84)
0.8-0.2	0.98	0.74	0.49	0.24	0.25	
1.0-0.0	0.99	0.99	0.49	0.49	0.50	
0.4-0.6	0.97	0.98	0.73	0.73	0.74	
0.2-0.8	0.96	0.97	0.98	0.99	0.99	
0.0-1.0	0.96	0.98	0.98	0.98	0.98	

Table G.4: Inner Loop Performances for p_m (PCM+NSGA-II, Rareness Level=1%)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	0.97	0.98	0.98	0.99	0.99	
0.05	0.98	0.99	0.74	0.74	0.75	
0.10	**0.98**	**0.99**	**0.99**	**0.99**	**0.99**	***(0.99)**
0.50	0.95	0.97	0.98	0.99	0.99	

G.1.2 Rareness Level = 10%

Table G.5 indicates the Fscore performances of inner cross validation for different values of *PopulationSize*, *GenerationSize* and *NumberOfGenerations*.

The hyper-parameter values whose inner cross validation performances are among the top four for all the given test folds are marked with **** in the last column. Due to the higher average value, the *PopulationSize*, *GenerationSize* and *NumberOfGenerations*

are set to 250, 200 and 10, respectively. Once these values are set, we repeat the same analysis for p_{rc} and p_{lc}.

The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column of Table G.6. Since it is distinguished with lower standard deviation, we set $p_{rc} = 0.6$ and $p_{lc} = 0.4$.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top one for all the given test folds are marked with * in the last column of Table G.7. Thus we set p_m=0.01.

Table G.5: Inner Loop Performances for *PopulationSize*, *GenerationSize* and *NumberOfGenerations* (PCM+NSGA-II, Rareness Level=10%)

PopulationSize	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5		**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5		**PopulationSize**	**GenerationSize**	**NumberOfGenerations**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5
150	**5**	**5**	0.92	0.86	0.91	0.88	0.87		**250**	**100**	**5**	0.95	0.93	0.93	0.95	0.92		**500**	**200**	**5**	0.92	0.92	0.92	0.94	0.91
150	**5**	**10**	0.92	0.86	0.91	0.88	0.87		**250**	**100**	**10**	0.93	0.91	0.93	0.93	0.90		**500**	**200**	**10**	0.92	0.92	0.91	0.93	0.90
150	**5**	**50**	0.88	0.94	0.99	0.96	0.96		**250**	**100**	**50**	0.90	0.91	0.94	0.94	0.91		**500**	**200**	**50**	0.86	0.88	0.87	0.89	0.85
150	**5**	**100**	0.90	0.94	0.99	0.96	0.96		**250**	**100**	**100**	0.90	0.91	0.94	0.94	0.91		**500**	**200**	**100**	0.82	0.83	0.83	0.85	0.85
150	**50**	**5**	0.92	0.92	0.92	0.94	0.90		**250**	**200**	**5**	0.93	0.94	0.95	0.94	0.94		**1000**	**5**	**5**	0.92	0.88	0.94	0.92	0.88
150	**50**	**10**	0.92	0.88	0.93	0.92	0.88		**250**	**200**	**10**	**0.94**	**0.95**	**0.96**	**0.95**	**0.96**	******(0.95)**	**1000**	**5**	**10**	0.92	0.87	0.91	0.89	0.87
150	**50**	**50**	0.92	0.91	0.94	0.94	0.91		**250**	**200**	**50**	0.94	0.93	0.92	0.95	0.92		**1000**	**5**	**50**	0.92	0.89	0.96	0.92	0.92
150	**50**	**100**	0.92	0.92	0.96	0.94	0.93	****(0.94)	**250**	**200**	**100**	0.90	0.88	0.88	0.90	0.86	****(0.94)	**1000**	**5**	**100**	0.94	0.86	0.96	0.90	0.90
150	**100**	**5**	0.92	0.92	0.93	0.95	0.92		**500**	**5**	**5**	0.92	0.92	0.97	0.96	0.94	****(0.94)	**1000**	**50**	**5**	0.92	0.88	0.96	0.94	0.90
150	**100**	**10**	0.93	0.93	0.92	0.94	0.91		**500**	**5**	**10**	0.94	0.92	0.98	0.96	0.94		**1000**	**50**	**10**	0.91	0.86	0.96	0.91	0.87
150	**100**	**50**	0.90	0.88	0.93	0.91	0.88		**500**	**5**	**50**	0.93	0.90	0.97	0.94	0.92		**1000**	**50**	**50**	0.85	0.90	0.97	0.94	0.92
150	**100**	**100**	0.86	0.90	0.93	0.91	0.88		**500**	**5**	**100**	0.94	0.84	0.98	0.88	0.87		**1000**	**50**	**100**	0.85	0.87	0.96	0.91	0.88
250	**5**	**5**	0.92	0.88	0.91	0.92	0.90		**500**	**50**	**5**	0.90	0.90	0.95	0.95	0.92		**1000**	**100**	**5**	0.88	0.88	0.92	0.90	0.87
250	**5**	**10**	0.95	0.88	0.91	0.92	0.90		**500**	**50**	**10**	0.90	0.90	0.93	0.94	0.90	****(0.94)	**1000**	**100**	**10**	0.85	0.90	0.92	0.90	0.87
250	**5**	**50**	0.94	0.88	0.92	0.92	0.90		**500**	**50**	**50**	0.92	0.92	0.98	0.96	0.94		**1000**	**100**	**50**	0.85	0.94	0.98	0.96	0.94
250	**5**	**100**	0.94	0.90	0.93	0.96	0.93		**500**	**50**	**100**	0.92	0.90	0.96	0.96	0.92		**1,000**	**100**	**100**	0.80	0.95	0.98	0.95	0.95
250	**50**	**5**	0.92	0.88	0.92	0.92	0.90		**500**	**100**	**5**	0.93	0.91	0.93	0.93	0.90		**1000**	**200**	**5**	0.91	0.90	0.90	0.90	0.88
250	**50**	**10**	0.92	0.90	0.94	0.94	0.91		**500**	**100**	**10**	0.93	0.89	0.93	0.91	0.88		**1000**	**200**	**10**	0.88	0.88	0.87	0.89	0.85
250	**50**	**50**	0.94	0.91	0.94	0.94	0.91		**500**	**100**	**50**	0.88	0.88	0.96	0.92	0.92		**1000**	**200**	**50**	0.84	0.84	0.87	0.86	0.84
250	**50**	**100**	0.92	0.93	0.98	0.96	0.94	****(0.94)	**500**	**100**	**100**	0.87	0.88	0.96	0.92	0.92		**1000**	**200**	**100**	0.84	0.85	0.91	0.88	0.86

Table G.6: Inner Loop Performances for p_{rc}, p_{lc} (PCM+NSGA-II, Rareness Level=10%)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	0.94	0.95	0.96	0.95	0.96	**(0.95)
0.6-0.4	**0.95**	**0.95**	**0.96**	**0.95**	**0.95**	**(0.95)
0.8-0.2	0.94	0.91	0.90	0.92	0.90	
1.0-0.0	0.95	0.92	0.94	0.96	0.93	
0.4-0.6	0.94	0.95	0.96	0.95	0.96	**(0.95)
0.2-0.8	0.93	0.92	0.93	0.92	0.94	
0.0-1.0	0.94	0.92	0.93	0.92	0.93	

Table G.7: Inner Loop Performances for p_m (PCM+NSGA-II, Rareness Level=10%)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.95**	**0.95**	**0.96**	**0.95**	**0.95**	***(0.95)**
0.05	0.92	0.89	0.91	0.91	0.89	
0.10	0.91	0.91	0.89	0.92	0.90	
0.50	0.83	0.85	0.84	0.86	0.86	

G.2 Hyper-parameter Optimization for PCM+RECGA

G.2.1 Rareness Level = 1%

Table G.8 indicates the Fscore performances of inner cross validation for different values of *PopulationSize* and *minFinalSetSize*.

The hyper-parameter values whose inner cross validation performances are among

the top one for all the given test folds are marked with * in the last column. We set *PopulationSize* and *minFinalSetSize* as 1000 and 50, respectively.

Once these values are set, we repeat the same analysis for p_{rc} and p_{lc}. The hyper-parameter values whose inner cross validation performances are among the top one for all the given test folds are marked with * in the last column of Table G.9. Then, $p_{rc} = 0.0$ and $p_{lc} = 1.0$.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top one for all the given test folds are marked with * in the last column of Table G.10. Thus we set p_m=0.01.

Table G.8: Inner Loop Performances for *PopulationSize* and *minFinalSetSize* (PCM+RECGA, Rareness Level=1%)

PopulationSize	**minFinalSetSize**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	0.94	0.93	0.93	0.94	0.94	
150	**50**	0.94	0.93	0.93	0.94	0.94	
150	**100**	0.94	0.93	0.93	0.94	0.94	
250	**5**	0.94	0.93	0.93	0.94	0.95	
250	**50**	0.94	0.93	0.93	0.94	0.95	
250	**100**	0.94	0.93	0.93	0.94	0.95	
250	**200**	0.94	0.93	0.93	0.94	0.95	
500	**5**	0.94	0.93	0.93	0.94	0.95	
500	**50**	0.94	0.93	0.93	0.94	0.95	
500	**100**	0.94	0.93	0.93	0.94	0.95	
500	**200**	0.94	0.93	0.93	0.94	0.95	
1000	**5**	0.94	0.93	0.94	0.95	0.95	*(0.94)
1000	**50**	**0.94**	**0.93**	**0.94**	**0.95**	**0.95**	***(0.94)**
1000	**100**	0.94	0.93	0.94	0.95	0.95	*(0.94)
1000	**200**	0.94	0.93	0.94	0.95	0.95	*(0.94)

Table G.9: Inner Loop Performances for p_{rc}, p_{lc} (PCM+RECGA, Rareness Level=1%)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	0.94	0.93	0.94	0.95	0.95	
0.6-0.4	0.70	0.93	0.94	0.95	0.95	
0.8-0.2	0.75	0.95	0.94	0.94	0.95	
1.0-0.0	0.75	0.95	0.94	0.95	0.95	
0.4-0.6	0.95	0.94	0.94	0.95	0.95	
0.2-0.8	0.96	0.95	0.96	0.96	0.96	
0.0-1.0	**0.98**	**0.98**	**0.98**	**0.98**	**0.99**	***(0.98)**

Table G.10: Inner Loop Performances for p_m (PCM+RECGA, Rareness Level=1%)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.98**	**0.98**	**0.98**	**0.98**	**0.99**	***(0.98)**
0.05	0.94	0.94	0.95	0.96	0.96	
0.10	0.93	0.93	0.94	0.94	0.95	
0.50	0.93	0.92	0.93	0.93	0.94	

G.2.2 Rareness Level = 10%

Table G.11 indicates the Fscore performances of inner cross validation for different values of *PopulationSize* and *minFinalSetSize*.

The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column. Due to the lower standard deviation, we set *PopulationSize* and *minFinalSetSize* as 250 and

200, respectively.

Once these values are set, we repeat the same analysis for p_{rc} and p_{lc}. The hyper-parameter values whose inner cross validation performances are among the top two for all the given test folds are marked with ** in the last column of Table G.12. Then, $p_{rc} = 0.5$ and $p_{lc} = 0.5$.

Finally, we tune the value of p_m by following the same procedure. The hyper-parameter value whose inner cross validation performance is among the top two for all the given test folds are marked with ** in the last column of Table G.13. Thus we set p_m=0.01.

Table G.11: Inner Loop Performances for *PopulationSize* and *minFinalSetSize* (PCM+RECGA, Rareness Level=10%)

PopulationSize	**minFinalSetSize**	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
150	**5**	0.96	0.93	0.95	0.96	0.94	
150	**50**	0.96	0.92	0.95	0.96	0.94	
150	**100**	0.96	0.92	0.95	0.96	0.94	
250	**5**	0.97	0.95	0.96	0.95	0.95	**(0.956)
250	**50**	0.97	0.94	0.96	0.96	0.95	**(0.956)
250	**100**	0.97	0.94	0.96	0.96	0.95	**(0.956)
250	**200**	**0.96**	**0.95**	**0.97**	**0.96**	**0.960**	****(0.96)**
500	**5**	0.97	0.95	0.97	0.95	0.96	**(0.960)
500	**50**	0.96	0.95	0.96	0.95	0.95	**(0.954)
500	**100**	0.96	0.95	0.96	0.95	0.95	**(0.954)
500	**200**	0.96	0.95	0.96	0.95	0.95	**(0.954)
1000	**5**	0.97	0.94	0.94	0.94	0.93	
1000	**50**	0.96	0.94	0.94	0.94	0.93	
1000	**100**	0.96	0.92	0.94	0.96	0.93	
1000	**200**	0.96	0.92	0.94	0.96	0.93	

Table G.12: Inner Loop Performances for p_{rc}, p_{lc} (PCM+RECGA, Rareness Level=10%)

prc-plc	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.5-0.5	**0.96**	**0.95**	**0.97**	**0.96**	**0.96**	****(0.960)**
0.6-0.4	0.97	0.95	0.96	0.95	0.96	**(0.958)
0.8-0.2	0.97	0.95	0.96	0.95	0.96	**(0.958)
1.0-0.0	0.94	0.89	0.92	0.94	0.93	
0.4-0.6	0.96	0.95	0.96	0.95	0.95	**(0.954)
0.2-0.8	0.95	0.95	0.97	0.96	0.96	
0.0-1.0	0.96	0.92	0.94	0.96	0.93	

Table G.13: Inner Loop Performances for p_m (PCM+RECGA, Rareness Level=1%)

pm	Test fold =#1	Test fold =#2	Test fold =#3	Test fold =#4	Test fold =#5	
0.01	**0.96**	**0.95**	**0.97**	**0.96**	**0.96**	****(0.960)**
0.05	0.95	0.93	0.95	0.96	0.94	**(0.946)
0.10	0.95	0.92	0.94	0.97	0.93	
0.50	0.95	0.93	0.95	0.96	0.94	**(0.946)

H Generalization Performances of the Models for the Wisconsin Breast Cancer Diagnostic Dataset: 5-fold Cross Validation

To see the generalization performance of the models with the selected hyper-parameter values, we evaluate the models in the outer loop of the nested cross validation. Thus, we test the model performances with 5-fold cross validation. We repeat the experiments for the rareness levels ranging between 1% to 37%.

Table H.1 shows the dataset configurations used in the experiments which are conducted with 5-fold CV.

Table H.1: Experimental Settings for the WBCD Dataset (5-fold CV)

	Malign	**Benign**	**Rareness level**		**Malign**	**Benign**	**Rareness level**
$\mathcal{S}$	2	140		$\mathcal{S}$	16	140	
$\mathcal{V}$	2	140		$\mathcal{V}$	16	140	
$\widetilde{\mathcal{S}}$	1	70	**1%**	$\widetilde{\mathcal{S}}$	8	70	**10%**
Total	5	350		Total	40	350	
$\mathcal{S}$	4	140		$\mathcal{S}$	24	140	
$\mathcal{V}$	4	140		$\mathcal{V}$	24	140	
$\widetilde{\mathcal{S}}$	2	70	**3%**	$\widetilde{\mathcal{S}}$	12	70	**15%**
Total	10	350		Total	60	350	
$\mathcal{S}$	8	140		$\mathcal{S}$	46	140	
$\mathcal{V}$	8	140		$\mathcal{V}$	46	140	
$\widetilde{\mathcal{S}}$	4	70	**5%**	$\widetilde{\mathcal{S}}$	23	70	**25%**
Total	20	350		Total	115	350	
$\mathcal{S}$	10	140		$\mathcal{S}$	82	140	
$\mathcal{V}$	10	140		$\mathcal{V}$	82	140	
$\widetilde{\mathcal{S}}$	5	70	**7%**	$\widetilde{\mathcal{S}}$	41	70	**37%**
Total	25	350		Total	205	350	
L			{0,...,140}				

Tables H.2 and H.3 summarize the average performance results and standard deviations of performance indicators of PCM+NSGA-II and PCM+RECGA. The perfor-

mances are given for both of the hyper-parameters determined for rareness levels 1% and 10%.

For the sake of completeness, we also report the performances of competitor models when the experiments are conducted with 5-fold cross validation. Their average training and test performances are given in Table H.4 and standard deviations of performance indicators for training and test are given in Table H.5.

Table H.2: Performances of PCM+NGSA-II (5-fold CV, WBCD Dataset)

PCM+NSGA-II	1%		3%		5%		7%		10%		15%		25%		37%	
	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
AVERAGE PERFORMANCE RESULTS																
$\mathcal{S}$																
Sensitivity	0.80	1.00	0.85	1.00	0.95	1.00	0.92	0.94	0.81	0.98	0.80	0.98	0.97	0.97	0.98	1.00
Specificity	0.96	0.78	0.99	0.85	0.99	0.92	0.98	0.89	0.99	0.97	0.98	0.99	0.98	0.97	0.98	0.98
Accuracy	0.96	0.78	0.98	0.85	0.99	0.92	0.98	0.89	0.97	0.97	0.95	0.98	0.97	0.97	0.98	0.99
Fscore	0.85	0.87	0.91	0.92	0.97	0.96	0.95	0.91	0.89	0.97	0.87	0.98	0.97	0.97	0.98	0.99
Fmeasure	0.37	0.13	0.73	0.28	0.88	0.59	0.85	0.55	0.86	0.87	0.83	0.95	0.95	0.94	0.97	0.98
$\mathcal{V}$																
Sensitivity	0.70	1.00	0.70	0.90	0.75	0.85	0.76	0.92	0.78	0.85	0.78	0.86	0.89	0.87	0.95	0.94
Specificity	0.96	0.76	0.98	0.86	0.98	0.87	0.96	0.88	0.99	0.91	0.97	0.93	0.95	0.95	0.97	0.95
Accuracy	0.96	0.76	0.97	0.86	0.96	0.87	0.95	0.88	0.96	0.91	0.94	0.92	0.93	0.93	0.96	0.95
Fscore	0.79	0.86	0.81	0.87	0.85	0.86	0.84	0.90	0.87	0.88	0.86	0.89	0.92	0.91	0.96	0.95
Fmeasure	0.36	0.12	0.63	0.26	0.69	0.42	0.66	0.51	0.82	0.65	0.80	0.76	0.87	0.86	0.95	0.93
$\tilde{\mathcal{S}}$																
Sensitivity	0.80	0.80	0.80	1.00	0.70	0.95	0.80	0.96	0.75	0.88	0.73	0.83	0.87	0.87	0.93	0.94
Specificity	0.93	0.76	0.99	0.85	0.98	0.90	0.97	0.89	0.98	0.91	0.95	0.93	0.95	0.93	0.96	0.96
Accuracy	0.93	0.76	0.98	0.85	0.97	0.90	0.96	0.89	0.96	0.91	0.92	0.91	0.93	0.92	0.95	0.95
Fscore	0.77	0.67	0.86	0.92	0.80	0.92	0.86	0.92	0.84	0.89	0.82	0.87	0.91	0.90	0.95	0.95
Fmeasure	0.29	0.08	0.73	0.29	0.70	0.53	0.71	0.56	0.78	0.67	0.73	0.74	0.86	0.84	0.93	0.94
STANDARD DEVIATIONS OF PERFORMANCE INDICATORS																
$\mathcal{S}$																
Sensitivity	0.00	0.24	0.00	0.12	0.00	0.06	0.05	0.04	0.03	0.09	0.03	0.17	0.03	0.03	0.00	0.03
Specificity	0.10	0.02	0.02	0.01	0.03	0.01	0.02	0.01	0.02	0.01	0.00	0.02	0.02	0.00	0.01	0.01
Accuracy	0.10	0.02	0.02	0.01	0.03	0.00	0.02	0.01	0.02	0.01	0.01	0.02	0.01	0.01	0.01	0.01
Fscore	0.06	0.15	0.01	0.07	0.02	0.03	0.03	0.02	0.02	0.05	0.02	0.10	0.02	0.02	0.01	0.02
Fmeasure	0.06	0.08	0.02	0.05	0.09	0.04	0.07	0.04	0.07	0.06	0.02	0.08	0.03	0.02	0.01	0.02
$\mathcal{V}$																
Sensitivity	0.00	0.24	0.12	0.10	0.05	0.08	0.07	0.10	0.05	0.05	0.07	0.08	0.03	0.03	0.01	0.01
Specificity	0.09	0.03	0.03	0.02	0.04	0.01	0.02	0.01	0.01	0.00	0.04	0.01	0.01	0.01	0.01	0.00
Accuracy	0.09	0.03	0.03	0.02	0.03	0.01	0.02	0.01	0.01	0.01	0.03	0.02	0.01	0.01	0.01	0.00
Fscore	0.06	0.15	0.05	0.07	0.03	0.05	0.04	0.06	0.03	0.03	0.03	0.05	0.02	0.02	0.01	0.00
Fmeasure	0.04	0.16	0.03	0.15	0.06	0.06	0.04	0.06	0.04	0.05	0.07	0.06	0.02	0.02	0.01	0.01
$\tilde{\mathcal{S}}$																
Sensitivity	0.40	0.40	0.00	0.24	0.10	0.19	0.08	0.18	0.08	0.14	0.12	0.14	0.05	0.05	0.04	0.05
Specificity	0.09	0.05	0.05	0.02	0.05	0.02	0.05	0.01	0.03	0.01	0.03	0.03	0.03	0.03	0.02	0.02
Accuracy	0.09	0.04	0.05	0.02	0.04	0.02	0.04	0.01	0.02	0.01	0.02	0.01	0.03	0.03	0.01	0.02
Fscore	0.34	0.38	0.03	0.16	0.03	0.13	0.03	0.11	0.04	0.09	0.06	0.09	0.04	0.04	0.02	0.02
Fmeasure	0.04	0.23	0.07	0.17	0.11	0.16	0.08	0.08	0.05	0.07	0.05	0.06	0.06	0.06	0.02	0.03

Table H.3: Performances of PCM+RECGA (5-fold CV, WBCD Dataset)

PCM+RECGA	1%		3%		5%		7%		10%		15%		25%		37%	
	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
AVERAGE PERFORMANCE RESULTS																
$\mathcal{S}$																
Sensitivity	1.00	1.00	1.00	1.00	0.93	0.93	0.90	0.90	0.93	0.89	0.98	0.98	0.98	0.98	1.00	1.00
Specificity	0.93	0.91	0.96	1.00	0.96	0.95	0.92	0.93	0.99	0.95	0.97	0.95	0.97	0.98	0.95	0.96
Accuracy	0.93	0.91	0.97	1.00	0.96	0.95	0.92	0.93	0.98	0.95	0.97	0.96	0.97	0.98	0.97	0.98
Fscore	0.96	0.95	0.98	1.00	0.94	0.93	0.91	0.91	0.95	0.92	0.98	0.97	0.97	0.98	0.98	0.98
Fmeasure	0.31	0.28	0.83	0.96	0.75	0.72	0.70	0.70	0.92	0.80	0.91	0.88	0.94	0.96	0.96	0.97
$\mathcal{V}$																
Sensitivity	1.00	1.00	1.00	0.90	0.83	0.85	0.80	0.82	0.84	0.90	0.87	0.86	0.90	0.90	0.97	0.96
Specificity	0.89	0.86	0.96	0.98	0.95	0.95	0.91	0.93	0.95	0.93	0.91	0.90	0.93	0.94	0.91	0.93
Accuracy	0.89	0.86	0.96	0.98	0.94	0.94	0.90	0.92	0.94	0.93	0.91	0.89	0.92	0.93	0.93	0.94
Fscore	0.94	0.93	0.98	0.93	0.88	0.90	0.84	0.86	0.89	0.91	0.89	0.88	0.92	0.92	0.94	0.95
Fmeasure	0.21	0.18	0.76	0.78	0.63	0.64	0.55	0.58	0.74	0.73	0.74	0.71	0.85	0.87	0.91	0.93
$\widetilde{\mathcal{S}}$																
Sensitivity	0.80	0.80	0.80	0.80	0.70	0.75	0.72	0.80	0.85	0.85	0.83	0.85	0.89	0.87	0.96	0.96
Specificity	0.88	0.87	0.96	0.99	0.97	0.96	0.92	0.92	0.95	0.91	0.90	0.89	0.91	0.94	0.93	0.95
Accuracy	0.88	0.86	0.96	0.99	0.95	0.95	0.91	0.91	0.94	0.91	0.89	0.89	0.91	0.92	0.94	0.95
Fscore	0.75	0.74	0.85	0.86	0.77	0.82	0.80	0.85	0.89	0.87	0.86	0.87	0.90	0.90	0.94	0.95
Fmeasure	0.16	0.15	0.68	0.75	0.59	0.63	0.56	0.58	0.75	0.70	0.69	0.69	0.83	0.85	0.92	0.94
STANDARD DEVIATIONS OF PERFORMANCE INDICATORS																
$\mathcal{S}$																
Sensitivity	0.00	0.00	0.00	0.00	0.10	0.10	0.06	0.06	0.08	0.05	0.02	0.02	0.02	0.02	0.00	0.00
Specificity	0.06	0.03	0.01	0.07	0.05	0.03	0.07	0.11	0.05	0.02	0.03	0.02	0.01	0.01	0.02	0.01
Accuracy	0.05	0.03	0.01	0.07	0.05	0.03	0.07	0.10	0.04	0.01	0.03	0.01	0.01	0.01	0.01	0.01
Fscore	0.03	0.02	0.00	0.04	0.05	0.05	0.05	0.07	0.04	0.02	0.02	0.01	0.01	0.01	0.01	0.01
Fmeasure	0.10	0.07	0.08	0.29	0.19	0.15	0.19	0.21	0.12	0.04	0.07	0.03	0.02	0.02	0.02	0.01
$\mathcal{V}$																
Sensitivity	0.00	0.00	0.20	0.00	0.05	0.06	0.15	0.13	0.08	0.08	0.06	0.06	0.02	0.01	0.01	0.01
Specificity	0.04	0.02	0.02	0.07	0.03	0.03	0.03	0.07	0.05	0.03	0.04	0.03	0.01	0.01	0.02	0.02
Accuracy	0.04	0.02	0.02	0.06	0.03	0.03	0.02	0.06	0.04	0.02	0.04	0.03	0.01	0.01	0.01	0.01
Fscore	0.02	0.01	0.13	0.04	0.04	0.04	0.08	0.07	0.03	0.04	0.04	0.03	0.01	0.01	0.01	0.01
Fmeasure	0.04	0.04	0.28	0.31	0.15	0.15	0.06	0.12	0.09	0.08	0.09	0.06	0.02	0.01	0.01	0.01
$\widetilde{\mathcal{S}}$																
Sensitivity	0.40	0.40	0.24	0.24	0.22	0.29	0.13	0.10	0.12	0.12	0.10	0.12	0.05	0.06	0.03	0.04
Specificity	0.04	0.02	0.01	0.07	0.03	0.03	0.06	0.08	0.09	0.05	0.04	0.03	0.02	0.05	0.03	0.02
Accuracy	0.04	0.02	0.01	0.07	0.03	0.02	0.05	0.07	0.07	0.04	0.03	0.03	0.03	0.04	0.02	0.02
Fscore	0.37	0.38	0.16	0.15	0.14	0.22	0.06	0.06	0.06	0.07	0.05	0.06	0.04	0.04	0.02	0.02
Fmeasure	0.09	0.09	0.17	0.30	0.12	0.17	0.11	0.15	0.15	0.11	0.07	0.07	0.05	0.07	0.03	0.02

Table H.4: Average Performance Results of Competitor Models (5-fold CV, WBCD Dataset)

	LR	pen-LR	SVM	ANN	DT	RF	LR	pen-LR	SVM	ANN	DT	RF
			$\mathcal{S} \cup \mathcal{V}$						$\widetilde{\mathcal{S}}$			
						1%						
Sensitivity	1.00	1.00	0.90	1.00	0.80	1.00	0.80	0.60	0.80	0.80	0.60	0.40
Specificity	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99
Accuracy	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	1.00	1.00	0.99	0.99
Fscore	1.00	1.00	0.94	1.00	0.89	1.00	0.80	0.60	0.80	0.80	0.60	0.40
Fmeasure	1.00	1.00	0.94	1.00	0.89	1.00	0.80	0.53	0.80	0.80	0.53	0.33
						3%						
Sensitivity	1.00	1.00	0.80	1.00	0.90	1.00	0.70	0.70	0.50	0.70	0.60	0.50
Specificity	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99
Accuracy	1.00	1.00	0.99	1.00	0.99	1.00	0.99	0.99	0.99	0.99	0.98	0.98
Fscore	1.00	1.00	0.80	1.00	0.94	1.00	0.73	0.73	0.53	0.73	0.60	0.60
Fmeasure	1.00	1.00	0.80	1.00	0.88	1.00	0.73	0.73	0.53	0.73	0.56	0.57
						5%						
Sensitivity	1.00	0.94	0.73	1.00	0.94	1.00	0.85	0.85	0.60	0.80	0.70	0.80
Specificity	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	1.00	0.99	0.97	0.98
Accuracy	1.00	1.00	0.99	1.00	1.00	1.00	0.98	0.98	0.98	0.98	0.95	0.97
Fscore	1.00	0.97	0.78	1.00	0.97	1.00	0.90	0.90	0.68	0.87	0.81	0.88
Fmeasure	1.00	0.97	0.78	1.00	0.96	1.00	0.81	0.84	0.68	0.80	0.65	0.75
						7%						
Sensitivity	1.00	0.96	0.89	1.00	0.91	1.00	0.76	0.76	0.76	0.80	0.72	0.72
Specificity	1.00	1.00	1.00	1.00	1.00	1.00	0.98	0.99	0.99	0.98	0.97	0.97
Accuracy	1.00	1.00	0.99	1.00	0.99	1.00	0.96	0.98	0.97	0.97	0.95	0.95
Fscore	1.00	0.98	0.94	1.00	0.95	1.00	0.83	0.84	0.84	0.86	0.81	0.80
Fmeasure	1.00	0.98	0.94	1.00	0.94	1.00	0.73	0.81	0.79	0.77	0.69	0.69
						10%						
Sensitivity	0.96	0.97	0.97	0.99	0.92	1.00	0.83	0.85	0.83	0.83	0.75	0.83
Specificity	1.00	1.00	1.00	1.00	0.99	1.00	0.98	0.99	1.00	0.98	0.96	0.99
Accuracy	0.99	1.00	1.00	1.00	0.99	1.00	0.96	0.98	0.98	0.96	0.94	0.97
Fscore	0.98	0.98	0.98	1.00	0.95	1.00	0.88	0.90	0.90	0.88	0.84	0.89
Fmeasure	0.97	0.98	0.98	1.00	0.93	1.00	0.82	0.87	0.89	0.82	0.74	0.85
						15%						
Sensitivity	0.96	0.96	0.97	0.99	0.95	1.00	0.90	0.90	0.92	0.90	0.77	0.85
Specificity	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	1.00	0.98	0.95	0.97
Accuracy	0.99	0.99	0.99	1.00	0.98	1.00	0.97	0.98	0.99	0.97	0.93	0.96
Fscore	0.98	0.98	0.98	1.00	0.97	1.00	0.94	0.94	0.96	0.94	0.85	0.91
Fmeasure	0.97	0.98	0.98	0.99	0.95	1.00	0.91	0.92	0.96	0.89	0.76	0.85
						25%						
Sensitivity	0.95	0.95	0.93	0.98	0.97	1.00	0.93	0.93	0.90	0.94	0.77	0.86
Specificity	0.99	0.99	1.00	1.00	0.99	1.00	0.98	0.99	0.99	0.96	0.97	0.99
Accuracy	0.98	0.98	0.98	0.99	0.99	1.00	0.97	0.97	0.97	0.96	0.92	0.96
Fscore	0.97	0.97	0.97	0.99	0.98	1.00	0.96	0.96	0.94	0.95	0.86	0.92
Fmeasure	0.96	0.97	0.96	0.99	0.97	1.00	0.94	0.94	0.93	0.92	0.83	0.91
						37%						
Sensitivity	0.97	0.97	0.97	0.98	0.98	1.00	0.96	0.96	0.95	0.95	0.91	0.93
Specificity	0.99	0.99	1.00	1.00	0.99	1.00	0.98	0.98	0.97	0.97	0.96	0.98
Accuracy	0.98	0.98	0.99	0.99	0.99	1.00	0.97	0.97	0.96	0.96	0.94	0.96
Fscore	0.98	0.98	0.98	0.99	0.99	1.00	0.97	0.97	0.96	0.96	0.93	0.95
Fmeasure	0.98	0.98	0.98	0.99	0.98	1.00	0.96	0.96	0.95	0.95	0.92	0.95

Table H.5: Standard Deviations of Performance Indicators of Competitor Models (5-fold CV, WBCD Dataset)

	LR	pen-LR	SVM	ANN	DT	RF	LR	pen-LR	SVM	ANN	DT	RF
			$\mathcal{S} \cup \mathcal{V}$						$\widetilde{\mathcal{S}}$			
						1%						
Sensitivity	0.00	0.00	0.12	0.00	0.10	0.00	0.40	0.49	0.40	0.40	0.49	0.49
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.01
Accuracy	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01
Fscore	0.00	0.00	0.07	0.00	0.06	0.00	0.40	0.49	0.40	0.40	0.49	0.49
Fmeasure	0.00	0.00	0.07	0.00	0.06	0.00	0.40	0.45	0.40	0.40	0.45	0.42
						3%						
Sensitivity	0.00	0.00	0.40	0.00	0.05	0.00	0.40	0.40	0.45	0.40	0.49	0.32
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01
Accuracy	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.02	0.01
Fscore	0.00	0.00	0.40	0.00	0.03	0.00	0.39	0.39	0.45	0.39	0.49	0.33
Fmeasure	0.00	0.00	0.40	0.00	0.05	0.00	0.39	0.39	0.45	0.39	0.46	0.33
						5%						
Sensitivity	0.00	0.04	0.34	0.00	0.04	0.00	0.20	0.20	0.34	0.19	0.10	0.10
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01	0.00	0.02	0.03	0.01
Accuracy	0.00	0.00	0.02	0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.01
Fscore	0.00	0.02	0.33	0.00	0.02	0.00	0.13	0.13	0.35	0.12	0.07	0.05
Fmeasure	0.00	0.02	0.33	0.00	0.03	0.00	0.12	0.14	0.35	0.14	0.11	0.06
						7%						
Sensitivity	0.00	0.02	0.10	0.00	0.06	0.00	0.23	0.23	0.23	0.22	0.20	0.24
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01	0.02	0.02	0.04	0.03
Accuracy	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.02	0.03	0.03	0.03	0.03
Fscore	0.00	0.01	0.06	0.00	0.03	0.00	0.16	0.16	0.16	0.16	0.14	0.16
Fmeasure	0.00	0.01	0.06	0.00	0.02	0.00	0.17	0.17	0.21	0.19	0.17	0.19
						10%						
Sensitivity	0.04	0.02	0.02	0.01	0.06	0.00	0.19	0.18	0.13	0.19	0.08	0.10
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01	0.01	0.02	0.03	0.02
Accuracy	0.01	0.00	0.00	0.00	0.01	0.00	0.03	0.02	0.01	0.03	0.04	0.02
Fscore	0.02	0.01	0.01	0.01	0.03	0.00	0.12	0.12	0.08	0.12	0.06	0.06
Fmeasure	0.03	0.01	0.01	0.01	0.05	0.00	0.14	0.11	0.07	0.14	0.12	0.08
						15%						
Sensitivity	0.02	0.01	0.01	0.02	0.03	0.00	0.06	0.06	0.05	0.06	0.08	0.08
Specificity	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.00	0.02	0.03	0.02
Accuracy	0.00	0.00	0.00	0.00	0.01	0.00	0.02	0.02	0.01	0.02	0.02	0.02
Fscore	0.01	0.01	0.01	0.01	0.02	0.00	0.03	0.03	0.03	0.04	0.04	0.04
Fmeasure	0.01	0.01	0.01	0.01	0.02	0.00	0.05	0.05	0.03	0.07	0.05	0.05
						25%						
Sensitivity	0.01	0.01	0.03	0.02	0.01	0.00	0.03	0.03	0.05	0.03	0.07	0.06
Specificity	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.02	0.02	0.02	0.02	0.01
Accuracy	0.00	0.00	0.01	0.01	0.00	0.00	0.01	0.02	0.03	0.02	0.02	0.02
Fscore	0.01	0.01	0.02	0.01	0.00	0.00	0.02	0.02	0.04	0.03	0.05	0.03
Fmeasure	0.01	0.01	0.02	0.02	0.01	0.00	0.03	0.04	0.05	0.05	0.05	0.04
						37%						
Sensitivity	0.01	0.01	0.01	0.01	0.00	0.00	0.02	0.02	0.03	0.03	0.03	0.03
Specificity	0.00	0.00	0.00	0.00	0.01	0.00	0.02	0.02	0.04	0.01	0.03	0.02
Accuracy	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.02	0.01	0.01	0.02
Fscore	0.00	0.00	0.01	0.01	0.00	0.00	0.01	0.01	0.02	0.01	0.01	0.02
Fmeasure	0.01	0.00	0.01	0.01	0.00	0.00	0.01	0.01	0.02	0.01	0.02	0.02

Note that the given performances of PCM+NSGA-II and PCM+RECGA are similar to those of the results of experimental analyses performed with randomly generated 100 instances, except for configurations where the rareness level is extremely low. For such configurations, since the samples consist of very few positive observations and the experiments are repeated only five times in 5-fold cross validation, the results are susceptible to these factors. For such configurations, repeating the experiments as much as possible reflects the model performances' better.

I Detailed Results of the Models Applied to the In-Stent-Restenosis Dataset

The average model performances in randomly generated 100 instances are given in the Tables I.1, I.2, I.3 and I.4 for Setting 1, and I.5, I.6, I.7 and I.8 for Setting 2.

Note that, PCM+NSGA-II and PCM+RECGA first utilize $\mathcal{S}$ to generate initial set of solutions, and then they tune these solutions with $\mathcal{V}$. On the other hand, since Random+NSGA-II and Random+RECGA generate the initial solutions randomly, they only use $\mathcal{V}$ for training. All the models' performances are tested in $\widetilde{\mathcal{S}}$.

Table I.1: Performances of PCM+NSGA-II (Setting 1)

	PCM+NSGA-II	$\mathcal{S}$		$\mathcal{V}$		$\widetilde{\mathcal{S}}$	
		AVG	STD.DEV	AVG	STD.DEV	AVG	STD.DEV
Number	True positive	14.02	3.21	15.43	2.29	6.97	2.10
	True negative	76.64	9.56	78.11	9.23	37.85	5.38
	True classification	90.66	10.08	93.54	10.55	44.82	5.65
Ratio	True positive	0.58	0.13	0.64	0.10	0.58	0.17
	True negative	0.80	0.10	0.81	0.10	0.79	0.11
	True classification	0.76	0.08	0.78	0.09	0.75	0.09
	Fscore	0.66	0.10	0.71	0.08	0.65	0.13
	Fmeasure	0.49	0.08	0.55	0.08	0.48	0.11
		Time (sec.)					
	AVG	14.3					
	STD.DEV	1.79					

Table I.2: Performances of Random+NSGA-II (Setting 1)

	Random+NSGA-II	$\mathcal{V}$		$\widetilde{\mathcal{S}}$	
		AVG	STD.DEV	AVG	STD.DEV
Number	True positive	6.36	2.85	2.90	1.75
	True negative	88.67	9.17	43.61	4.88
	True classification	95.03	9.96	46.51	5.13
Ratio	True positive	0.26	0.12	0.24	0.15
	True negative	0.92	0.10	0.91	0.10
	True classification	0.79	0.08	0.78	0.09
	Fscore	0.40	0.15	0.36	0.18
	Fmeasure	0.33	0.13	0.29	0.14
		Time (sec.)			
	AVG	6.75			
	STD.DEV	0.16			

Table I.3: Performances of PCM+RECGA (Setting 1)

	PCM+RECGA	$\mathcal{S}$		$\mathcal{V}$		$\widetilde{\mathcal{S}}$	
		AVG	STD.DEV	AVG	STD.DEV	AVG	STD.DEV
Number	True positive	19.17	2.34	17.92	1.99	8.48	1.83
	True negative	67.97	6.57	67.70	6.37	32.64	4.15
	True classification	87.14	5.62	85.62	5.73	41.12	4.04
Ratio	True positive	0.80	0.10	0.75	0.08	0.71	0.15
	True negative	0.71	0.07	0.71	0.07	0.68	0.09
	True classification	0.73	0.05	0.71	0.05	0.69	0.07
	Fscore	0.74	0.04	0.72	0.04	0.68	0.08
	Fmeasure	0.54	0.05	0.51	0.05	0.47	0.09
		Time (sec.)					
	AVG	8.25					
	STD.DEV	1.86					

Table I.4: Performances of Random+RECGA (Setting 1)

Random+RECGA		$\mathcal{V}$		$\widetilde{\mathcal{S}}$	
		AVG	STD.DEV	AVG	STD.DEV
Number	True positive	11.14	2.91	5.37	2.16
	True negative	80.18	7.69	39.19	4.97
	True classification	91.32	6.25	44.57	4.40
Ratio	True positive	0.46	0.12	0.45	0.18
	True negative	0.84	0.08	0.82	0.10
	True classification	0.76	0.05	0.74	0.07
	Fscore	0.58	0.10	0.55	0.16
	Fmeasure	0.43	0.08	0.40	0.13
		Time (sec.)			
AVG		0.83			
STD.DEV		0.31			

Table I.5: Performances of PCM+NSGA-II (Setting 2)

PCM+NSGA-II		$\mathcal{S}$		$\mathcal{V}$		$\widetilde{\mathcal{S}}$	
		AVG	STD.DEV	AVG	STD.DEV	AVG	STD.DEV
Number	True positive	15.18	4.32	15.81	3.86	7.54	2.37
	True negative	15.96	4.48	16.83	4.00	7.94	2.40
	True classification	31.14	7.15	32.64	7.26	15.48	3.80
Ratio	True positive	0.63	0.18	0.66	0.16	0.63	0.20
	True negative	0.67	0.19	0.70	0.17	0.66	0.20
	True classification	0.65	0.15	0.68	0.15	0.65	0.16
	Fscore	0.63	0.15	0.67	0.15	0.62	0.16
	Fmeasure	0.64	0.15	0.67	0.15	0.63	0.17
Time (sec.)							
AVG		103.65					
STD.DEV		3.02					

Table I.6: Performances of Random+NSGA-II (Setting 2)

Random+NSGA-II		$\mathcal{V}$		$\widetilde{\mathcal{S}}$	
		AVG	STD.DEV	AVG	STD.DEV
Number	True positive	10.93	3.51	5.27	2.33
	True negative	18.95	4.21	9.14	2.47
	True classification	29.88	6.76	14.41	3.71
Ratio	True positive	0.46	0.15	0.44	0.19
	True negative	0.79	0.18	0.76	0.21
	True classification	0.62	0.14	0.60	0.15
	Fscore	0.57	0.15	0.53	0.18
	Fmeasure	0.54	0.15	0.51	0.18
Time (sec.)					
AVG		104.3			
STD.DEV		3.71			

Table I.7: Performances of PCM+RECGA (Setting 2)

	PCM+RECGA	$\mathcal{S}$		$\mathcal{V}$		$\widetilde{\mathcal{S}}$	
		AVG	STD.DEV	AVG	STD.DEV	AVG	STD.DEV
Number	True positive	18.48	2.79	18.39	2.20	8.63	1.76
	True negative	15.60	3.41	16.60	2.27	7.30	1.88
	True classification	34.08	3.58	34.99	2.65	15.93	2.36
Ratio	True positive	0.77	0.12	0.77	0.09	0.72	0.15
	True negative	0.65	0.14	0.69	0.09	0.61	0.16
	True classification	0.71	0.07	0.73	0.06	0.66	0.10
	Fscore	0.69	0.09	0.72	0.06	0.64	0.11
	Fmeasure	0.72	0.07	0.74	0.06	0.68	0.10
		Time (sec.)					
	AVG	0.86					
	STD.DEV	0.35					

Table I.8: Performances of Random+RECGA (Setting 2)

	Random+RECGA	$\mathcal{V}$		$\widetilde{\mathcal{S}}$	
		AVG	STD.DEV	AVG	STD.DEV
Number	True positive	8.98	4.69	4.69	2.50
	True negative	20.90	2.65	10.20	1.66
	True classification	29.88	3.63	14.89	2.00
Ratio	True positive	0.37	0.20	0.39	0.21
	True negative	0.87	0.11	0.85	0.14
	True classification	0.62	0.08	0.62	0.08
	Fscore	0.48	0.19	0.49	0.20
	Fmeasure	0.47	0.19	0.47	0.20
		Time (sec.)			
	AVG	0.23			
	STD.DEV	0.06			

J Predictor Values and Real Restenosis Status of 100 Patients in Test Sample

Table J.1: Predictor Values and Real Restenosis Status of 100 Patients in Test Sample

F1	F2	F3	F4	F5	F6	F7	F8	Real status of restenosis
0	0	0	0	2.75	0	1	0	0
1	1	0	0	3	0	1	0	1
1	1	0	1	3	0	0	0	0
1	1	0	1	3.1	1	1	0	1
1	0	0	1	2	0	1	0	1
1	1	1	1	3.5	0	1	0	1
1	0	1	0	2.5	0	0	0	1
1	0	0	1	3	0	0	0	0
1	0	0	0	2.75	0	0	0	1
1	0	0	0	3	0	0	0	1
0	0	0	0	2.75	0	0	0	0
1	0	0	0	3.5	0	0	0	0
1	0	0	0	3	0	0	0	0
1	0	0	0	2.75	0	1	0	1
0	0	0	0	3.5	0	1	0	0
1	0	0	0	2.75	0	1	1	1
1	0	0	0	4.5	0	1	0	1
1	0	0	1	4	0	0	1	1
0	0	0	0	3	0	0	0	0
0	0	0	0	3	0	1	0	0
0	0	0	0	3	0	0	0	0
1	0	0	0	3	0	1	0	0
0	0	0	0	2.9	0	0	0	0
0	0	0	0	3.5	0	1	0	1
1	0	0	0	3	0	1	0	1
1	0	0	0	2.75	0	0	0	1
1	0	0	0	2.75	1	0	0	1

0	0	0	0	2.75	0	0	0	0
1	0	0	0	4	0	1	0	1
1	0	1	1	3.5	0	0	1	1
1	0	0	0	2.75	1	0	0	1
1	0	0	1	2.75	0	0	1	1
0	0	1	1	3	0	0	0	1
1	0	0	0	4	0	0	0	1
1	0	0	0	3.5	0	1	0	1
0	0	0	0	2.75	0	0	0	0
1	0	0	0	3.5	0	1	0	0
1	0	0	0	3	0	0	0	1
0	0	0	0	2.75	0	1	0	0
1	0	0	0	3.5	0	0	0	0
1	0	0	0	3.5	0	1	0	0
1	0	0	0	3.5	0	1	0	1
0	0	0	0	2.75	0	0	0	0
1	0	0	0	3.5	0	1	0	0
0	0	0	0	2.75	0	0	0	0
1	0	0	0	3.5	0	0	0	1
1	0	0	0	3.5	0	0	0	0
0	0	0	0	2.75	0	0	0	0
1	0	0	0	3.5	0	1	0	0
1	1	1	0	3	0	0	0	1
0	1	0	0	2.5	0	1	0	1
1	0	0	1	3.5	0	0	0	0
0	0	0	0	2.75	0	0	0	0
0	0	0	0	2.75	0	1	0	0
0	0	0	0	3	0	1	0	0
1	0	0	0	3	0	0	0	0
1	0	0	0	3	0	0	0	0
1	0	0	0	3	0	0	0	1
1	0	0	0	2.75	0	1	0	0

1	0	0	0	2.75	0	0	0	0
1	0	0	0	3.5	0	1	0	0
0	0	0	0	3	0	1	0	0
1	0	0	0	2.75	0	0	0	0
1	0	0	0	4	0	0	0	0
1	1	0	0	3	1	0	0	1
0	0	0	1	3	0	1	0	1
1	0	0	0	2.75	0	1	0	1
0	0	0	0	2.75	0	1	0	0
1	0	0	0	3	0	0	0	1
1	0	0	1	3	0	0	0	0
0	0	0	0	3	0	0	0	0
1	0	0	0	3	0	1	0	0
1	0	0	0	2.75	0	1	0	0
1	0	0	0	3.5	0	1	0	0
0	0	0	0	2.75	0	1	0	0
1	0	0	0	4	0	1	0	0
1	0	0	0	3	0	1	0	0
1	0	0	0	2.75	0	1	0	0
1	0	1	0	2.75	0	1	0	1
1	0	0	0	3	1	0	0	1
1	0	0	0	4	0	1	0	1
1	0	0	1	3	0	1	1	1
1	0	0	0	2.75	0	1	0	1
1	0	0	1	2.75	0	1	0	1
0	0	0	1	3	0	1	0	1
1	0	0	0	3	0	1	0	1
1	0	0	0	2.75	0	1	0	1
0	1	1	0	3	1	1	0	1
1	0	0	0	2.75	0	1	0	1
1	0	0	0	3	0	0	0	1
1	0	0	0	3	0	1	0	1

1	0	0	0	2.75	0	0	0	1
1	1	0	0	2.75	0	1	0	1
1	0	0	1	2.75	0	1	0	1
0	0	0	0	2.5	0	1	0	1
0	0	0	0	2.5	0	1	0	0
1	0	1	0	3	0	1	0	1
1	0	0	0	3.5	0	1	0	0
0	0	0	0	2.75	0	1	0	0
1	0	0	0	3.5	0	1	0	0

K Detailed Results of the Models Applied to the Wisconsin Breast Cancer Original Dataset

The average model performances in randomly generated 100 instances are given in the Tables K.1, K.2, K.3, K.4, K.5, K.6, K.7 and K.8.

Table K.1: Average Performance Results of PCM+NSGA-II (WBCO Dataset)

	PCM+NSGA-II	1%		3%		5%		7%		10%		15%		25%		35%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
	$\mathcal{S}$																
Number	True positive	1.95	1.51	4.63	3.75	6.76	6.53	9.35	9.36	14.54	14.66	22.96	22.65	46.22	44.56	75.65	73.15
Number	True negative	136.09	144.69	142.10	146.12	145.02	145.07	145.33	145.35	145.67	145.45	145.10	145.13	143.78	144.19	143.29	143.41
Number	True classification	138.04	146.20	146.73	149.87	151.78	151.60	154.68	154.71	160.21	160.11	168.06	167.78	190.00	188.75	218.94	216.56
Ratio	True positive	0.98	0.76	0.93	0.75	0.85	0.82	0.85	0.85	0.86	0.86	0.88	0.87	0.94	0.91	0.96	0.93
Ratio	True negative	0.92	0.98	0.96	0.99	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.97	0.97	0.97	0.97
Ratio	True classification	0.92	0.97	0.96	0.98	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.96	0.96	0.96	0.96	0.95
	Fscore	0.94	0.81	0.94	0.83	0.90	0.87	0.90	0.90	0.91	0.91	0.93	0.92	0.96	0.94	0.96	0.94
	Fmeasure	0.31	0.58	0.64	0.71	0.77	0.75	0.81	0.81	0.86	0.86	0.88	0.88	0.93	0.91	0.95	0.93
	$\mathcal{V}$																
Number	True positive	1.86	1.40	4.32	4.10	6.18	6.75	8.77	9.68	14.21	15.38	23.09	23.91	45.90	45.55	75.69	74.26
Number	True negative	138.28	145.69	142.73	147.29	145.49	146.52	145.87	146.85	145.87	146.76	145.73	146.43	145.10	146.11	145.01	145.44
Number	True classification	140.14	147.09	147.05	151.39	151.67	153.27	154.64	156.53	160.08	162.14	168.82	170.34	191.00	191.66	220.70	219.70
Ratio	True positive	0.93	0.70	0.86	0.82	0.77	0.84	0.80	0.88	0.84	0.90	0.89	0.92	0.94	0.93	0.96	0.94
Ratio	True negative	0.93	0.98	0.96	1.00	0.98	0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.98	0.99	0.98	0.98
Ratio	True classification	0.93	0.98	0.96	0.99	0.97	0.98	0.97	0.98	0.97	0.98	0.97	0.98	0.97	0.97	0.97	0.97
	Fscore	0.92	0.79	0.90	0.89	0.86	0.91	0.88	0.93	0.90	0.94	0.93	0.95	0.96	0.96	0.97	0.96
	Fmeasure	0.31	0.66	0.62	0.84	0.74	0.84	0.80	0.89	0.85	0.91	0.90	0.93	0.94	0.94	0.96	0.95
	$\tilde{\mathcal{S}}$																
Number	True positive	1.85	1.30	4.27	3.36	6.58	6.10	8.55	8.34	13.23	13.52	21.91	21.76	45.37	44.29	74.63	72.14
Number	True negative	137.14	144.93	142.38	145.94	145.23	144.97	145.44	145.22	145.38	145.02	144.77	144.81	143.61	143.86	143.16	143.11
Number	True classification	138.99	146.23	146.65	149.30	151.81	151.07	153.99	153.56	158.61	158.54	166.68	166.57	188.98	188.15	217.79	215.25
Ratio	True positive	0.93	0.65	0.85	0.67	0.82	0.76	0.78	0.76	0.78	0.80	0.84	0.84	0.93	0.90	0.94	0.91
Ratio	True negative	0.93	0.98	0.96	0.99	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.97	0.97	0.97	0.97
Ratio	True classification	0.93	0.97	0.96	0.98	0.97	0.97	0.97	0.97	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.95
	Fscore	0.91	0.73	0.89	0.77	0.89	0.84	0.86	0.84	0.86	0.87	0.90	0.90	0.95	0.93	0.96	0.94
	Fmeasure	0.31	0.54	0.61	0.64	0.76	0.72	0.77	0.75	0.80	0.80	0.86	0.85	0.92	0.91	0.94	0.92
	Time (sec.)																
		12.37	8.80	12.44	8.89	12.73	9.08	13.05	13.14	15.17	58.75	16.36	67.18	18.57	69.49	21.32	73.63

Table K.2: Standard Deviations of Performance Indicators of PCM+NSGA-II (WBCO Dataset)

PCM+NSGA-II		1%		3%		5%		7%		10%		15%		25%		35%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
$\mathcal{S}$																	
Number	True positive	0.22	0.62	0.69	1.21	1.31	1.65	1.56	1.70	1.82	1.96	2.53	2.62	2.42	4.40	3.09	5.75
	True negative	7.98	14.64	4.97	1.81	2.96	3.32	2.37	2.04	1.55	2.07	2.05	2.27	2.28	2.56	2.45	3.78
	True classification	7.93	14.79	4.77	2.27	3.02	3.55	2.30	2.20	2.08	2.45	2.96	2.90	2.72	3.92	3.05	5.34
Ratio	True positive	0.11	0.31	0.14	0.24	0.16	0.21	0.14	0.15	0.11	0.12	0.10	0.10	0.05	0.09	0.04	0.07
	True negative	0.05	0.10	0.03	0.01	0.02	0.02	0.02	0.01	0.01	0.01	0.01	0.02	0.02	0.02	0.02	0.03
	True classification	0.05	0.10	0.03	0.01	0.02	0.02	0.01	0.01	0.01	0.01	0.02	0.02	0.01	0.02	0.01	0.02
	Fscore	0.07	0.27	0.08	0.20	0.11	0.14	0.09	0.10	0.06	0.07	0.06	0.06	0.02	0.05	0.02	0.04
	Fmeasure	0.16	0.26	0.15	0.20	0.13	0.16	0.10	0.10	0.07	0.07	0.06	0.06	0.03	0.05	0.02	0.04
$\mathcal{V}$																	
Number	True positive	0.35	0.51	0.72	0.56	1.12	0.91	1.11	0.87	1.38	1.26	1.50	1.94	1.24	3.30	1.44	5.15
	True negative	5.27	14.67	3.84	0.70	1.38	3.23	1.13	0.88	0.90	1.10	0.86	1.17	1.04	1.41	1.06	3.99
	True classification	5.20	14.81	3.59	0.92	1.65	3.38	1.59	1.32	1.77	1.83	1.71	2.47	1.94	3.70	2.22	6.56
Ratio	True positive	0.17	0.25	0.14	0.11	0.14	0.11	0.10	0.08	0.08	0.07	0.06	0.07	0.03	0.07	0.02	0.07
	True negative	0.04	0.10	0.03	0.00	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.03
	True classification	0.03	0.10	0.02	0.01	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.01	0.03
	Fscore	0.11	0.18	0.08	0.07	0.09	0.07	0.06	0.05	0.05	0.04	0.03	0.05	0.02	0.04	0.01	0.04
	Fmeasure	0.14	0.19	0.12	0.09	0.10	0.11	0.07	0.06	0.06	0.06	0.04	0.05	0.02	0.04	0.01	0.04
$\tilde{\mathcal{S}}$																	
Number	True positive	0.38	0.66	0.95	1.15	1.13	1.41	1.64	1.77	2.11	2.33	2.39	2.64	2.36	3.34	3.22	6.67
	True negative	7.36	14.65	4.92	1.86	2.33	3.69	1.76	2.05	2.08	2.22	2.09	2.18	2.35	2.66	2.33	4.29
	True classification	7.25	14.78	4.58	1.86	2.13	3.55	1.87	2.28	2.31	2.60	2.60	2.77	2.71	3.07	3.25	6.49
Ratio	True positive	0.19	0.33	0.19	0.23	0.14	0.18	0.15	0.16	0.12	0.14	0.09	0.10	0.05	0.07	0.04	0.08
	True negative	0.05	0.10	0.03	0.01	0.02	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.02	0.02	0.03
	True classification	0.05	0.10	0.03	0.01	0.01	0.02	0.01	0.01	0.01	0.02	0.01	0.02	0.01	0.02	0.01	0.03
	Fscore	0.14	0.30	0.12	0.19	0.09	0.12	0.10	0.11	0.08	0.09	0.05	0.06	0.02	0.04	0.02	0.05
	Fmeasure	0.16	0.26	0.14	0.18	0.10	0.13	0.09	0.11	0.08	0.09	0.06	0.06	0.03	0.03	0.02	0.05
Time (sec.)																	
		0.51	0.61	0.25	0.27	0.27	0.32	0.55	14.13	0.73	19.17	0.33	2.20	0.48	2.31	0.63	6.53

Table K.3: Average Performance Results of Random+NSGA-II (WBCO Dataset)

Random+NSGA-II		1%		3%		5%		7%		10%		15%		25%		35%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
								$\mathcal{V}$									
Number	True positive	1.75	1.62	3.99	4.04	5.31	6.04	7.37	8.70	10.83	13.43	16.17	20.44	29.76	37.75	51.97	61.66
	True negative	140.85	145.79	144.78	147.00	145.72	147.04	146.20	147.21	146.97	147.35	147.28	147.43	147.44	147.36	147.37	147.21
	True classification	142.60	147.41	148.77	151.04	151.03	153.08	153.57	155.91	157.80	160.78	163.45	167.87	177.20	185.11	199.34	208.87
Ratio	True positive	0.88	0.81	0.80	0.81	0.66	0.76	0.67	0.79	0.64	0.79	0.62	0.79	0.61	0.77	0.66	0.78
	True negative	0.95	0.99	0.98	0.99	0.98	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	0.99
	True classification	0.95	0.98	0.97	0.99	0.97	0.98	0.97	0.98	0.96	0.97	0.94	0.96	0.90	0.94	0.88	0.92
	Fscore	0.89	0.87	0.87	0.88	0.78	0.85	0.79	0.87	0.77	0.88	0.76	0.87	0.75	0.86	0.79	0.87
	Fmeasure	0.38	0.59	0.67	0.81	0.68	0.80	0.72	0.84	0.74	0.86	0.75	0.87	0.75	0.86	0.79	0.87
								$\tilde{\mathcal{S}}$									
Number	True positive	1.61	1.46	4.05	3.71	6.30	6.03	7.51	7.72	9.45	10.81	14.78	17.62	30.92	36.05	48.00	57.54
	True negative	139.91	144.73	144.51	146.24	145.75	146.34	146.16	146.65	146.86	147.07	147.31	147.19	147.34	146.93	147.36	146.67
	True classification	141.52	146.19	148.56	149.95	152.05	152.37	153.67	154.37	156.31	157.88	162.09	164.81	178.26	182.98	195.36	204.21
Ratio	True positive	0.81	0.73	0.81	0.74	0.79	0.75	0.68	0.70	0.56	0.64	0.57	0.68	0.63	0.74	0.61	0.73
	True negative	0.95	0.98	0.98	0.99	0.98	0.99	0.99	0.99	0.99	0.99	1.00	0.99	1.00	0.99	1.00	0.99
	True classification	0.94	0.97	0.97	0.98	0.97	0.98	0.97	0.97	0.95	0.96	0.93	0.95	0.90	0.93	0.86	0.90
	Fscore	0.83	0.80	0.87	0.83	0.87	0.85	0.79	0.81	0.70	0.77	0.72	0.80	0.77	0.84	0.75	0.83
	Fmeasure	0.34	0.46	0.67	0.71	0.77	0.77	0.73	0.76	0.67	0.74	0.71	0.79	0.76	0.83	0.75	0.83
								Time (sec.)									
		8.64	4.28	8.69	4.36	8.87	4.46	9.01	4.58	9.28	4.79	9.82	5.11	11.07	5.85	12.54	6.78

Table K.4: Standard Deviations of Performance Indicators of Random+NSGA-II (WBCO Dataset)

Random+NSGA-II		1%		3%		5%		7%		10%		15%		25%		35%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
$\mathcal{V}$																	
Number	True positive	0.49	0.46	0.68	0.83	1.22	1.38	1.43	1.67	1.83	2.06	2.78	2.93	5.76	4.25	8.40	5.12
	True negative	1.77	5.42	0.99	2.55	0.89	1.70	0.82	1.54	0.80	1.20	0.65	0.90	0.62	0.61	0.80	0.56
	True classification	1.82	5.27	1.17	2.47	1.34	1.58	1.55	1.39	1.99	1.84	2.80	2.61	5.88	4.17	8.54	4.96
Ratio	True positive	0.24	0.23	0.14	0.17	0.15	0.17	0.13	0.15	0.11	0.12	0.11	0.11	0.12	0.09	0.11	0.06
	True negative	0.01	0.04	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.00	0.00	0.01	0.00
	True classification	0.01	0.04	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.01	0.03	0.02	0.04	0.02
	Fscore	0.16	0.16	0.09	0.11	0.11	0.13	0.09	0.11	0.07	0.09	0.07	0.09	0.09	0.07	0.07	0.05
	Fmeasure	0.21	0.17	0.11	0.13	0.11	0.11	0.09	0.09	0.07	0.08	0.07	0.08	0.09	0.07	0.07	0.05
$\tilde{\mathcal{S}}$																	
Number	True positive	0.59	0.56	0.98	0.89	1.24	1.19	1.74	1.96	2.38	2.88	2.78	3.06	5.75	5.55	9.77	6.67
	True negative	2.06	7.00	1.64	3.49	1.51	2.40	1.33	1.71	1.21	1.50	0.89	0.92	1.12	0.71	1.58	0.73
	True classification	2.13	6.79	1.71	3.32	1.63	2.42	1.70	2.06	2.35	2.45	2.77	2.94	5.49	5.45	9.28	6.61
Ratio	True positive	0.30	0.28	0.20	0.18	0.15	0.15	0.16	0.18	0.14	0.17	0.11	0.12	0.12	0.11	0.12	0.08
	True negative	0.01	0.05	0.01	0.02	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.00
	True classification	0.01	0.05	0.01	0.02	0.01	0.02	0.01	0.01	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.03
	Fscore	0.24	0.22	0.15	0.11	0.11	0.10	0.11	0.13	0.11	0.14	0.08	0.10	0.09	0.09	0.09	0.07
	Fmeasure	0.23	0.19	0.15	0.14	0.11	0.12	0.10	0.12	0.11	0.13	0.08	0.10	0.09	0.09	0.09	0.07
Time (sec.)																	
		0.17	0.22	0.09	0.17	0.09	0.13	0.07	0.16	0.09	0.18	0.11	0.15	0.09	0.14	0.11	0.15

Table K.5: Average Performance Results of PCM+RECGA (WBCO Dataset)

	PCM+RECGA	1%		3%		5%		7%		10%		15%		25%		35%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
	$\mathcal{S}$																
Number	True positive	1.77	1.71	4.53	4.44	7.32	7.24	9.97	9.82	15.06	15.52	22.66	23.05	43.38	43.68	73.63	73.73
Number	True negative	141.29	142.11	143.29	143.58	142.00	142.47	142.37	142.17	143.47	142.69	142.66	143.22	142.45	142.62	140.91	140.89
Number	True classification	143.06	143.82	147.82	148.02	149.32	149.71	152.34	151.99	158.53	158.21	165.32	166.27	185.83	186.30	214.54	214.62
Ratio	True positive	0.89	0.86	0.91	0.89	0.92	0.91	0.91	0.89	0.89	0.91	0.87	0.89	0.89	0.89	0.93	0.93
Ratio	True negative	0.95	0.96	0.97	0.97	0.96	0.96	0.96	0.96	0.97	0.96	0.96	0.97	0.96	0.96	0.95	0.95
Ratio	True classification	0.95	0.96	0.97	0.97	0.96	0.96	0.96	0.96	0.96	0.96	0.95	0.96	0.94	0.95	0.95	0.95
	Fscore	0.88	0.87	0.92	0.91	0.93	0.92	0.92	0.91	0.91	0.93	0.90	0.91	0.91	0.91	0.93	0.94
	Fmeasure	0.47	0.47	0.69	0.68	0.73	0.73	0.77	0.75	0.82	0.82	0.84	0.85	0.88	0.88	0.92	0.92
	$\mathcal{V}$																
Number	True positive	1.83	1.86	4.32	4.46	6.58	6.85	9.39	9.64	14.24	14.79	22.33	22.65	41.82	42.10	71.49	71.77
Number	True negative	141.92	143.20	143.54	144.01	142.39	143.10	142.81	142.56	143.70	143.28	142.54	143.08	143.08	143.32	141.98	141.87
Number	True classification	143.75	145.06	147.86	148.47	148.97	149.95	152.20	152.20	157.94	158.07	164.87	165.73	184.90	185.42	213.47	213.64
Ratio	True positive	0.92	0.93	0.86	0.89	0.82	0.86	0.85	0.88	0.84	0.87	0.86	0.87	0.85	0.86	0.90	0.91
Ratio	True negative	0.96	0.97	0.97	0.97	0.96	0.97	0.96	0.96	0.97	0.97	0.96	0.97	0.97	0.97	0.96	0.96
Ratio	True classification	0.96	0.97	0.97	0.97	0.95	0.96	0.96	0.96	0.96	0.96	0.95	0.95	0.94	0.94	0.94	0.94
	Fscore	0.92	0.94	0.90	0.92	0.88	0.90	0.90	0.91	0.89	0.91	0.90	0.91	0.89	0.90	0.92	0.93
	Fmeasure	0.56	0.59	0.68	0.70	0.69	0.72	0.75	0.76	0.80	0.81	0.83	0.84	0.86	0.87	0.91	0.91
	$\tilde{\mathcal{S}}$																
Number	True positive	1.47	1.45	3.76	3.82	6.35	6.42	8.75	9.00	13.38	13.82	21.39	21.70	42.10	42.37	71.71	71.88
Number	True negative	141.18	142.54	143.17	143.34	141.64	142.14	141.91	141.62	143.18	142.39	141.81	142.38	141.78	142.04	140.38	140.27
Number	True classification	142.65	143.99	146.93	147.16	147.99	148.56	150.66	150.62	156.56	156.21	163.20	164.08	183.88	184.41	212.09	212.15
Ratio	True positive	0.74	0.73	0.75	0.76	0.79	0.80	0.80	0.82	0.79	0.81	0.82	0.83	0.86	0.86	0.91	0.91
Ratio	True negative	0.95	0.96	0.97	0.97	0.96	0.96	0.96	0.96	0.97	0.96	0.96	0.96	0.96	0.96	0.95	0.95
Ratio	True classification	0.95	0.96	0.96	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.94	0.94	0.93	0.94	0.93	0.93
	Fscore	0.76	0.76	0.82	0.83	0.85	0.86	0.85	0.87	0.85	0.87	0.87	0.88	0.89	0.90	0.92	0.92
	Fmeasure	0.40	0.42	0.59	0.60	0.65	0.66	0.69	0.70	0.76	0.76	0.80	0.81	0.86	0.86	0.90	0.90
	Time (sec.)																
		4.59	7.87	4.57	6.52	4.76	6.39	4.97	10.53	6.55	56.39	7.26	64.78	8.47	66.94	10.06	69.25

Table K.6: Standard Deviations of Performance Indicators of PCM+RECGA (WBCO Dataset)

PCM+RECGA		1%		3%		5%		7%		10%		15%		25%		35%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
								$\mathcal{S}$									
Number	True positive	0.53	0.53	0.95	0.89	1.24	1.13	1.81	1.72	2.57	2.86	4.90	5.26	8.87	9.01	10.31	10.23
	True negative	7.66	9.98	4.37	5.19	7.45	8.41	6.23	6.25	4.71	4.68	3.81	9.79	4.64	6.44	5.63	5.79
	True classification	7.57	9.89	4.31	5.16	7.37	8.25	5.82	5.92	4.61	4.63	4.59	9.91	7.71	8.75	9.64	9.64
Ratio	True positive	0.27	0.26	0.19	0.18	0.16	0.14	0.16	0.16	0.15	0.17	0.19	0.20	0.18	0.18	0.13	0.13
	True negative	0.05	0.07	0.03	0.04	0.05	0.06	0.04	0.04	0.03	0.03	0.03	0.07	0.03	0.04	0.04	0.04
	True classification	0.05	0.07	0.03	0.03	0.05	0.05	0.04	0.04	0.03	0.03	0.03	0.06	0.04	0.04	0.04	0.04
	Fscore	0.22	0.23	0.13	0.13	0.10	0.09	0.11	0.10	0.10	0.11	0.13	0.14	0.12	0.12	0.08	0.08
	Fmeasure	0.25	0.25	0.19	0.20	0.15	0.16	0.13	0.13	0.11	0.12	0.11	0.14	0.11	0.11	0.08	0.08
								$\mathcal{V}$									
Number	True positive	0.35	0.38	0.73	0.79	1.11	1.23	1.58	1.67	2.32	2.60	4.15	4.32	8.87	8.94	9.92	9.86
	True negative	7.60	10.89	4.20	5.15	6.22	7.34	5.92	5.57	4.52	4.59	3.77	8.97	4.15	5.96	4.84	4.91
	True classification	7.56	10.84	4.04	4.96	5.92	6.91	5.55	4.99	4.28	4.27	4.13	8.77	8.19	8.84	9.18	9.11
Ratio	True positive	0.17	0.19	0.15	0.16	0.14	0.15	0.14	0.15	0.14	0.15	0.16	0.17	0.18	0.18	0.13	0.12
	True negative	0.05	0.07	0.03	0.03	0.04	0.05	0.04	0.04	0.03	0.03	0.03	0.06	0.03	0.04	0.03	0.03
	True classification	0.05	0.07	0.03	0.03	0.04	0.04	0.03	0.03	0.03	0.03	0.02	0.05	0.04	0.04	0.04	0.04
	Fscore	0.11	0.12	0.08	0.09	0.08	0.09	0.09	0.09	0.09	0.10	0.10	0.11	0.12	0.12	0.08	0.08
	Fmeasure	0.28	0.28	0.16	0.17	0.12	0.13	0.13	0.12	0.10	0.10	0.09	0.11	0.12	0.12	0.08	0.08
								$\tilde{\mathcal{S}}$									
Number	True positive	0.68	0.68	1.17	1.15	1.40	1.51	1.99	2.04	2.98	3.26	4.09	4.28	8.31	8.30	10.82	10.80
	True negative	7.89	11.63	4.91	5.58	7.30	8.40	6.71	6.44	4.92	4.86	4.63	9.54	4.37	5.86	5.82	5.94
	True classification	7.79	11.49	4.58	5.25	6.93	7.97	6.13	5.82	4.43	4.49	4.64	9.11	7.06	7.65	9.93	9.95
Ratio	True positive	0.34	0.34	0.23	0.23	0.18	0.19	0.18	0.19	0.18	0.19	0.16	0.16	0.17	0.17	0.14	0.14
	True negative	0.05	0.08	0.03	0.04	0.05	0.06	0.05	0.04	0.03	0.03	0.03	0.06	0.03	0.04	0.04	0.04
	True classification	0.05	0.08	0.03	0.03	0.04	0.05	0.04	0.04	0.03	0.03	0.03	0.05	0.04	0.04	0.04	0.04
	Fscore	0.31	0.31	0.18	0.18	0.12	0.13	0.12	0.12	0.12	0.13	0.10	0.11	0.11	0.11	0.09	0.09
	Fmeasure	0.26	0.25	0.17	0.18	0.13	0.15	0.13	0.13	0.11	0.12	0.09	0.11	0.10	0.10	0.09	0.09
								Time (sec.)									
		1.31	0.42	0.70	0.20	0.36	0.23	14.42	0.47	19.28	0.67	1.94	0.25	1.64	0.31	1.90	0.50

Table K.7: Average Performance Results of Random+RECGA (WBCO Dataset)

Random+RECGA		1%		3%		5%		7%		10%		15%		25%		35%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
								$\mathcal{V}$									
Number	True positive	1.73	1.73	4.05	4.05	5.62	5.62	7.90	7.90	11.58	11.58	17.73	17.73	30.47	30.47	50.85	50.85
	True negative	145.85	145.85	146.53	146.53	147.22	147.22	147.53	147.53	147.69	147.69	147.74	147.74	147.74	147.74	147.74	147.74
	True classification	147.58	147.58	150.58	150.58	152.84	152.84	155.43	155.43	159.27	159.27	165.47	165.47	178.21	178.21	198.59	198.59
Ratio	True positive	0.87	0.87	0.81	0.81	0.70	0.70	0.72	0.72	0.68	0.68	0.68	0.68	0.62	0.62	0.64	0.64
	True negative	0.99	0.99	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	True classification	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.97	0.97	0.95	0.95	0.90	0.90	0.87	0.87
	Fscore	0.90	0.90	0.88	0.88	0.81	0.81	0.83	0.83	0.80	0.80	0.81	0.81	0.76	0.76	0.78	0.78
	Fmeasure	0.62	0.62	0.77	0.77	0.77	0.77	0.81	0.81	0.80	0.80	0.80	0.80	0.76	0.76	0.78	0.78
								$\widetilde{\mathcal{S}}$									
Number	True positive	1.68	1.68	4.12	4.12	6.19	6.19	7.56	7.56	10.52	10.52	16.75	16.75	33.50	33.50	49.12	49.12
	True negative	144.82	144.82	146.04	146.04	146.91	146.91	147.33	147.33	147.67	147.67	147.70	147.70	147.70	147.70	147.70	147.70
	True classification	146.50	146.50	150.16	150.16	153.10	153.10	154.89	154.89	158.19	158.19	164.45	164.45	181.20	181.20	196.82	196.82
Ratio	True positive	0.84	0.84	0.82	0.82	0.77	0.77	0.69	0.69	0.62	0.62	0.64	0.64	0.68	0.68	0.62	0.62
	True negative	0.98	0.98	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	True classification	0.98	0.98	0.98	0.98	0.98	0.98	0.97	0.97	0.96	0.96	0.95	0.95	0.92	0.92	0.87	0.87
	Fscore	0.88	0.88	0.89	0.89	0.86	0.86	0.80	0.80	0.76	0.76	0.78	0.78	0.81	0.81	0.76	0.76
	Fmeasure	0.51	0.51	0.75	0.75	0.81	0.81	0.78	0.78	0.75	0.75	0.77	0.77	0.81	0.81	0.76	0.76
								Time (sec.)									
		0.99	1.61	1.03	1.64	1.10	1.77	1.11	1.80	1.18	1.92	1.25	2.06	1.47	2.44	1.75	2.90

Table K.8: Standard Deviations of Performance Indicators of Random+RECGA (WBCO Dataset)

Random+RECGA		1%		3%		5%		7%		10%		15%		25%		35%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
$\mathcal{V}$																	
Number	True positive	0.44	0.44	0.74	0.74	1.32	1.32	1.58	1.58	2.07	2.07	2.35	2.35	3.60	3.60	4.03	4.03
	True negative	1.58	1.58	1.24	1.24	0.84	0.84	0.64	0.64	0.52	0.52	0.44	0.44	0.44	0.44	0.44	0.44
	True classification	1.59	1.59	1.36	1.36	1.43	1.43	1.58	1.58	2.08	2.08	2.44	2.44	3.67	3.67	4.06	4.06
Ratio	True positive	0.22	0.22	0.15	0.15	0.16	0.16	0.14	0.14	0.12	0.12	0.09	0.09	0.07	0.07	0.05	0.05
	True negative	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	True classification	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.02	0.02	0.02
	Fscore	0.15	0.15	0.09	0.09	0.12	0.12	0.10	0.10	0.09	0.09	0.07	0.07	0.06	0.06	0.04	0.04
	Fmeasure	0.20	0.20	0.12	0.12	0.12	0.12	0.10	0.10	0.09	0.09	0.07	0.07	0.06	0.06	0.04	0.04
$\widetilde{\mathcal{S}}$																	
Number	True positive	0.51	0.51	0.88	0.88	1.10	1.10	1.65	1.65	2.07	2.07	2.66	2.66	3.06	3.06	4.24	4.24
	True negative	1.78	1.78	1.59	1.59	1.15	1.15	0.95	0.95	0.49	0.49	0.46	0.46	0.46	0.46	0.46	0.46
	True classification	1.81	1.81	1.66	1.66	1.39	1.39	1.65	1.65	2.11	2.11	2.70	2.70	3.12	3.12	4.28	4.28
Ratio	True positive	0.25	0.25	0.18	0.18	0.14	0.14	0.15	0.15	0.12	0.12	0.10	0.10	0.06	0.06	0.05	0.05
	True negative	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	True classification	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.02	0.02	0.02	0.02	0.02
	Fscore	0.19	0.19	0.12	0.12	0.09	0.09	0.11	0.11	0.10	0.10	0.08	0.08	0.04	0.04	0.04	0.04
	Fmeasure	0.18	0.18	0.13	0.13	0.10	0.10	0.11	0.11	0.10	0.10	0.08	0.08	0.04	0.04	0.04	0.04
Time (sec.)																	
		0.11	0.07	0.06	0.08	0.09	0.07	0.06	0.03	0.07	0.04	0.07	0.04	0.06	0.04	0.07	0.04

L Detailed Results of the Models Applied to the Wisconsin Breast Cancer Diagnostic Dataset

The average model performances in randomly generated 100 instances are given in the Tables L.1, L.2, L.3, L.4, L.5, L.6, L.7 and L.8.

Table L.1: Average Performance Results of PCM+NSGA-II (WBCD Dataset)

PCM+NSGA-II		1%		3%		5%		7%		10%		15%		25%		37%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
$\mathcal{S}$																	
Number	True positive	0.97	1.00	3.78	3.93	5.57	5.97	7.98	8.84	11.89	12.85	19.63	20.51	38.44	39.13	68.03	68.49
	True negative	91.58	65.97	113.57	95.88	115.62	100.59	116.47	104.97	116.68	108.81	116.78	112.85	116.36	114.16	116.60	115.64
	True classification	92.55	66.97	117.35	99.81	121.19	106.56	124.45	113.81	128.57	121.66	136.41	133.36	154.80	153.29	184.63	184.13
Ratio	True positive	0.97	1.00	0.95	0.98	0.93	1.00	0.89	0.98	0.91	0.99	0.93	0.98	0.96	0.98	0.97	0.98
	True negative	0.77	0.55	0.95	0.81	0.97	0.85	0.98	0.88	0.98	0.91	0.98	0.95	0.98	0.96	0.98	0.97
	True classification	0.77	0.56	0.95	0.81	0.97	0.85	0.97	0.89	0.97	0.92	0.97	0.95	0.97	0.96	0.98	0.97
	Fscore	0.83	0.71	0.94	0.88	0.94	0.91	0.92	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.98	0.97
	Fmeasure	0.10	0.04	0.63	0.28	0.77	0.42	0.82	0.59	0.88	0.73	0.92	0.87	0.95	0.93	0.97	0.97
$\mathcal{V}$																	
Number	True positive	0.89	1.00	3.34	3.76	4.83	5.54	6.87	8.13	10.63	11.81	17.84	18.22	36.23	35.53	65.50	64.62
	True negative	92.23	63.74	112.56	93.26	114.73	97.37	115.98	102.65	116.27	105.76	116.27	110.65	114.94	111.63	113.95	112.04
	True classification	93.12	64.74	115.90	97.02	119.56	102.91	122.85	110.78	126.90	117.57	134.11	128.87	151.17	147.16	179.45	176.66
Ratio	True positive	0.89	1.00	0.84	0.94	0.81	0.92	0.76	0.90	0.82	0.91	0.85	0.87	0.91	0.89	0.94	0.92
	True negative	0.78	0.54	0.95	0.78	0.96	0.82	0.97	0.86	0.98	0.89	0.98	0.93	0.97	0.94	0.96	0.94
	True classification	0.78	0.54	0.94	0.79	0.96	0.82	0.96	0.87	0.96	0.89	0.96	0.92	0.95	0.93	0.95	0.93
	Fscore	0.77	0.69	0.88	0.85	0.87	0.86	0.85	0.88	0.89	0.89	0.91	0.90	0.93	0.91	0.95	0.93
	Fmeasure	0.08	0.04	0.52	0.24	0.66	0.35	0.73	0.51	0.81	0.63	0.86	0.77	0.90	0.86	0.93	0.91
$\tilde{\mathcal{S}}$																	
Number	True positive	0.94	0.99	3.16	3.66	4.63	5.65	6.87	8.30	9.74	11.54	17.10	18.36	35.29	35.39	64.14	64.01
	True negative	93.41	66.75	113.41	96.55	115.21	100.13	116.05	103.89	116.07	107.10	115.67	110.89	114.06	112.20	113.29	112.67
	True classification	94.35	67.74	116.57	100.21	119.84	105.78	122.92	112.19	125.81	118.64	132.77	129.25	149.35	147.59	177.43	176.68
Ratio	True positive	0.94	0.99	0.79	0.92	0.77	0.94	0.76	0.92	0.75	0.89	0.81	0.87	0.88	0.88	0.92	0.91
	True negative	0.78	0.56	0.95	0.81	0.97	0.84	0.98	0.87	0.98	0.90	0.97	0.93	0.96	0.94	0.95	0.95
	True classification	0.79	0.56	0.95	0.81	0.96	0.85	0.96	0.88	0.95	0.90	0.95	0.92	0.94	0.93	0.94	0.93
	Fscore	0.82	0.71	0.85	0.85	0.84	0.88	0.84	0.89	0.84	0.89	0.88	0.90	0.92	0.91	0.93	0.93
	Fmeasure	0.10	0.04	0.54	0.26	0.65	0.39	0.73	0.54	0.76	0.65	0.83	0.78	0.88	0.86	0.92	0.91
Time (sec.)																	
		59.64	84.31	61.97	113.70	63.48	89.07	66.34	91.93	67.00	96.44	74.46	107.15	103.45	142.66	144.35	203.53

Table L.2: Standard Deviations of Performance Indicators of PCM+NSGA-II (WBCD Dataset)

	PCM+NSGA-II	1%		3%		5%		7%		10%		15%		25%		37%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
	$\mathcal{S}$																
Number	True positive	0.00	0.17	0.26	0.50	0.22	0.72	0.39	1.46	0.46	1.32	0.94	1.76	1.93	2.31	2.82	2.83
Number	True negative	9.80	16.57	9.44	5.31	7.30	2.99	8.78	2.80	5.83	2.03	4.88	1.77	3.35	1.72	2.35	2.07
Number	True classification	9.80	16.55	9.46	5.36	7.33	3.06	8.75	2.99	5.73	2.20	4.77	2.15	3.50	2.53	3.30	3.24
Ratio	True positive	0.00	0.17	0.06	0.13	0.04	0.12	0.04	0.16	0.04	0.10	0.04	0.08	0.05	0.06	0.04	0.04
Ratio	True negative	0.08	0.14	0.08	0.04	0.06	0.03	0.07	0.02	0.05	0.02	0.04	0.01	0.03	0.01	0.02	0.02
Ratio	True classification	0.08	0.14	0.08	0.04	0.06	0.02	0.07	0.02	0.04	0.02	0.03	0.02	0.02	0.02	0.02	0.02
	Fscore	0.07	0.18	0.06	0.08	0.04	0.07	0.05	0.11	0.03	0.06	0.03	0.05	0.03	0.03	0.02	0.02
	Fmeasure	0.01	0.09	0.09	0.18	0.11	0.15	0.14	0.13	0.10	0.07	0.07	0.05	0.04	0.03	0.02	0.02
	$\mathcal{V}$																
Number	True positive	0.00	0.31	0.47	0.62	0.59	0.81	0.80	0.98	1.09	1.09	1.44	0.91	1.62	1.21	1.75	1.43
Number	True negative	8.62	15.15	9.55	5.05	6.88	3.19	7.49	2.44	6.14	2.38	4.11	2.05	3.32	1.90	3.09	2.32
Number	True classification	8.62	15.11	9.55	5.07	6.86	3.20	7.43	2.55	5.81	2.49	4.02	2.37	3.47	2.53	3.49	2.93
Ratio	True positive	0.00	0.31	0.12	0.16	0.10	0.14	0.09	0.11	0.08	0.08	0.07	0.04	0.04	0.03	0.03	0.02
Ratio	True negative	0.07	0.13	0.08	0.04	0.06	0.03	0.06	0.02	0.05	0.02	0.03	0.02	0.03	0.02	0.03	0.02
Ratio	True classification	0.07	0.13	0.08	0.04	0.05	0.03	0.06	0.02	0.04	0.02	0.03	0.02	0.02	0.02	0.02	0.02
	Fscore	0.06	0.28	0.07	0.09	0.06	0.09	0.05	0.07	0.04	0.05	0.04	0.03	0.02	0.02	0.02	0.02
	Fmeasure	0.01	0.06	0.08	0.14	0.08	0.14	0.10	0.10	0.09	0.08	0.07	0.05	0.04	0.03	0.02	0.02
	$\tilde{\mathcal{S}}$																
Number	True positive	0.10	0.24	0.53	0.77	0.62	1.24	0.84	1.54	1.36	1.64	2.06	2.00	2.38	2.08	2.69	2.76
Number	True negative	9.71	15.66	8.79	4.88	7.11	2.74	7.84	3.03	5.83	2.36	4.65	2.44	3.59	2.90	3.54	3.26
Number	True classification	9.71	15.65	8.66	4.94	7.08	2.58	7.60	2.73	5.60	2.68	4.46	2.75	3.48	3.22	3.85	3.67
Ratio	True positive	0.10	0.24	0.13	0.19	0.10	0.21	0.09	0.17	0.10	0.13	0.10	0.10	0.06	0.05	0.04	0.04
Ratio	True negative	0.08	0.13	0.07	0.04	0.06	0.02	0.07	0.03	0.05	0.02	0.04	0.02	0.03	0.02	0.03	0.03
Ratio	True classification	0.08	0.13	0.07	0.04	0.06	0.02	0.06	0.02	0.04	0.02	0.03	0.02	0.02	0.02	0.02	0.02
	Fscore	0.10	0.22	0.07	0.13	0.06	0.15	0.05	0.12	0.06	0.08	0.05	0.06	0.03	0.03	0.02	0.02
	Fmeasure	0.01	0.10	0.08	0.18	0.09	0.14	0.12	0.12	0.10	0.10	0.08	0.07	0.04	0.04	0.03	0.03
	Time (sec.)																
		7.07	1.75	6.29	0.77	1.39	1.00	1.53	6.71	2.14	1.68	2.81	2.49	8.17	10.94	13.88	10.04

Table L.3: Average Performance Results of Random+NSGA-II (WBCD Dataset)

Random+NSGA-II		1%		3%		5%		7%		10%		15%		25%		37%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
$\mathcal{V}$																	
Number	True positive	0.99	1.00	3.79	3.92	5.69	5.81	8.55	8.75	12.45	12.66	20.22	20.51	38.46	39.11	67.65	68.46
	True negative	59.75	56.78	63.09	58.63	66.72	60.71	70.77	63.75	74.48	66.74	81.67	71.63	92.14	81.34	101.87	92.12
	True classification	60.74	57.78	66.88	62.55	72.41	66.52	79.32	72.50	86.93	79.40	101.89	92.14	130.60	120.45	169.52	160.58
Ratio	True positive	0.99	1.00	0.95	0.98	0.95	0.97	0.95	0.97	0.96	0.97	0.96	0.98	0.96	0.98	0.97	0.98
	True negative	0.50	0.48	0.53	0.49	0.56	0.51	0.59	0.54	0.63	0.56	0.69	0.60	0.77	0.68	0.86	0.77
	True classification	0.51	0.48	0.54	0.51	0.58	0.53	0.62	0.57	0.66	0.60	0.73	0.66	0.82	0.76	0.90	0.85
	Fscore	0.66	0.64	0.67	0.65	0.70	0.67	0.73	0.69	0.76	0.71	0.80	0.74	0.86	0.80	0.91	0.86
	Fmeasure	0.03	0.03	0.12	0.12	0.18	0.17	0.26	0.24	0.36	0.33	0.52	0.46	0.73	0.67	0.87	0.83
$\widetilde{\mathcal{S}}$																	
Number	True positive	0.98	0.99	3.78	3.93	5.71	5.92	8.52	8.88	12.13	12.66	20.10	20.61	38.76	39.52	66.70	68.12
	True negative	61.91	59.24	64.68	61.48	67.70	64.08	71.45	66.62	74.86	69.86	82.11	75.32	92.70	83.66	101.47	93.65
	True classification	62.89	60.23	68.46	65.41	73.41	70.00	79.97	75.50	86.99	82.52	102.21	95.93	131.46	123.18	168.17	161.77
Ratio	True positive	0.98	0.99	0.95	0.98	0.95	0.99	0.95	0.99	0.93	0.97	0.96	0.98	0.97	0.99	0.95	0.97
	True negative	0.52	0.50	0.54	0.52	0.57	0.54	0.60	0.56	0.63	0.59	0.69	0.63	0.78	0.70	0.85	0.79
	True classification	0.52	0.50	0.56	0.53	0.59	0.56	0.62	0.59	0.66	0.63	0.73	0.69	0.83	0.77	0.89	0.86
	Fscore	0.67	0.66	0.69	0.67	0.71	0.69	0.73	0.71	0.75	0.73	0.80	0.77	0.86	0.82	0.90	0.87
	Fmeasure	0.03	0.03	0.12	0.12	0.18	0.18	0.26	0.25	0.35	0.34	0.52	0.49	0.74	0.69	0.87	0.83
Time (sec.)																	
		12.22	38.24	13.10	40.36	13.65	41.29	14.23	43.40	15.13	46.29	16.89	53.04	21.45	70.18	30.00	104.17

Table L.4: Standard Deviations of Performance Indicators of Random+NSGA-II (WBCD Dataset)

Random+NSGA-II		1%		3%		5%		7%		10%		15%		25%		37%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
								$\mathcal{V}$									
Number	True positive	0.00	0.10	0.27	0.43	0.42	0.54	0.46	0.55	0.51	0.57	0.61	0.72	0.82	0.91	1.06	1.20
	True negative	6.28	7.92	6.07	7.71	6.64	7.32	7.28	6.54	6.63	6.37	6.77	5.67	6.25	5.71	5.47	3.42
	True classification	6.28	7.92	6.09	7.58	6.71	7.27	7.24	6.38	6.58	6.34	6.80	5.59	6.19	5.84	5.54	3.65
Ratio	True positive	0.00	0.10	0.07	0.11	0.07	0.09	0.05	0.06	0.04	0.04	0.03	0.03	0.02	0.02	0.02	0.02
	True negative	0.05	0.07	0.05	0.06	0.06	0.06	0.06	0.05	0.06	0.05	0.06	0.05	0.05	0.05	0.05	0.03
	True classification	0.05	0.07	0.05	0.06	0.05	0.06	0.06	0.05	0.05	0.05	0.05	0.04	0.04	0.04	0.03	0.02
	Fscore	0.05	0.09	0.05	0.05	0.05	0.05	0.05	0.04	0.05	0.04	0.05	0.03	0.04	0.03	0.03	0.02
	Fmeasure	0.00	0.01	0.01	0.02	0.02	0.02	0.03	0.03	0.03	0.04	0.04	0.04	0.04	0.04	0.03	0.02
								$\widetilde{\mathcal{S}}$									
Number	True positive	0.10	0.14	0.26	0.44	0.27	0.59	0.32	0.77	0.59	0.88	0.58	1.11	0.74	1.23	1.33	1.71
	True negative	6.00	7.32	6.09	6.88	6.51	6.82	6.68	6.62	6.33	6.63	5.66	5.72	5.25	5.94	4.93	4.63
	True classification	5.99	7.31	6.10	6.86	6.47	6.82	6.65	6.67	6.39	6.63	5.63	5.90	5.43	6.14	5.06	4.79
Ratio	True positive	0.10	0.14	0.06	0.11	0.05	0.10	0.04	0.09	0.05	0.07	0.03	0.05	0.02	0.03	0.02	0.02
	True negative	0.05	0.06	0.05	0.06	0.05	0.06	0.06	0.06	0.05	0.06	0.05	0.05	0.04	0.05	0.04	0.04
	True classification	0.05	0.06	0.05	0.06	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.04	0.03	0.04	0.03	0.03
	Fscore	0.08	0.11	0.05	0.06	0.05	0.06	0.05	0.05	0.04	0.04	0.04	0.04	0.03	0.03	0.03	0.02
	Fmeasure	0.00	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.03	0.04	0.03	0.05	0.03	0.04	0.03	0.03
								Time (sec.)									
		1.53	0.45	1.51	0.47	1.41	0.40	1.36	0.36	1.37	0.37	1.71	0.39	1.77	0.42	5.56	0.59

Table L.5: Average Performance Results of PCM+RECGA (WBCD Dataset)

	PCM+RECGA	1%		3%		5%		7%		10%		15%		25%		37%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
	$\mathcal{S}$																
Number	True positive	0.99	1.00	3.79	3.74	5.81	5.75	8.85	8.92	12.87	12.73	20.81	20.57	39.83	39.61	69.92	69.76
Number	True negative	106.42	107.60	110.91	112.77	111.49	112.82	110.88	111.72	111.92	112.32	113.37	113.94	113.18	114.23	113.02	114.86
Number	True classification	107.41	108.60	114.70	116.51	117.30	118.57	119.73	120.64	124.79	125.05	134.18	134.51	153.01	153.84	182.94	184.62
Ratio	True positive	0.99	1.00	0.95	0.94	0.97	0.96	0.98	0.99	0.99	0.98	0.99	0.98	1.00	0.99	1.00	1.00
Ratio	True negative	0.89	0.90	0.93	0.95	0.94	0.95	0.93	0.94	0.94	0.94	0.95	0.96	0.95	0.96	0.95	0.97
Ratio	True classification	0.90	0.90	0.93	0.95	0.94	0.95	0.94	0.94	0.95	0.95	0.96	0.96	0.96	0.97	0.97	0.98
	Fscore	0.93	0.94	0.93	0.93	0.95	0.95	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.98
	Fmeasure	0.45	0.46	0.59	0.63	0.69	0.72	0.73	0.75	0.80	0.81	0.88	0.89	0.93	0.94	0.96	0.97
	$\mathcal{V}$																
Number	True positive	1.00	1.00	3.87	3.79	5.64	5.55	8.21	8.02	11.80	11.84	18.82	19.05	36.81	36.72	66.54	66.12
Number	True negative	105.58	106.32	109.92	111.56	110.45	111.99	109.48	110.41	110.76	111.17	110.59	111.30	110.27	111.06	107.75	109.86
Number	True classification	106.58	107.32	113.79	115.35	116.09	117.54	117.69	118.43	122.56	123.01	129.41	130.35	147.08	147.78	174.29	175.98
Ratio	True positive	1.00	1.00	0.97	0.95	0.94	0.93	0.91	0.89	0.91	0.91	0.90	0.91	0.92	0.92	0.95	0.94
Ratio	True negative	0.89	0.89	0.92	0.94	0.93	0.94	0.92	0.93	0.93	0.93	0.93	0.94	0.93	0.93	0.91	0.92
Ratio	True classification	0.89	0.89	0.93	0.94	0.93	0.94	0.92	0.93	0.93	0.93	0.92	0.93	0.93	0.93	0.92	0.93
	Fscore	0.93	0.94	0.94	0.94	0.93	0.93	0.91	0.90	0.92	0.92	0.91	0.92	0.92	0.92	0.93	0.93
	Fmeasure	0.37	0.39	0.56	0.59	0.65	0.69	0.65	0.66	0.73	0.75	0.78	0.80	0.86	0.87	0.90	0.91
	$\tilde{\mathcal{S}}$																
Number	True positive	0.77	0.72	3.05	2.99	4.80	4.62	7.62	7.56	10.81	10.70	18.30	18.31	36.43	36.16	66.04	65.33
Number	True negative	105.20	106.15	109.54	111.54	110.64	111.86	109.86	110.59	110.63	110.94	110.45	111.27	110.16	111.13	108.46	110.17
Number	True classification	105.97	106.87	112.59	114.53	115.44	116.48	117.48	118.15	121.44	121.64	128.75	129.58	146.59	147.29	174.50	175.50
Ratio	True positive	0.77	0.72	0.76	0.75	0.80	0.77	0.85	0.84	0.83	0.82	0.87	0.87	0.91	0.90	0.94	0.93
Ratio	True negative	0.88	0.89	0.92	0.94	0.93	0.94	0.92	0.93	0.93	0.93	0.93	0.94	0.93	0.93	0.91	0.93
Ratio	True classification	0.88	0.89	0.92	0.93	0.92	0.93	0.92	0.92	0.92	0.92	0.92	0.93	0.92	0.93	0.92	0.93
	Fscore	0.70	0.67	0.80	0.80	0.84	0.82	0.87	0.87	0.87	0.87	0.90	0.90	0.92	0.92	0.93	0.93
	Fmeasure	0.24	0.26	0.43	0.46	0.56	0.57	0.62	0.63	0.68	0.69	0.77	0.78	0.86	0.86	0.90	0.91
	Time (sec.)																
		86.47	89.73	96.40	90.01	89.29	91.25	92.60	93.47	95.87	99.32	102.98	108.84	132.02	150.12	180.83	209.12

Table L.6: Standard Deviations of Performance Indicators of PCM+RECGA (WBCD Dataset)

	PCM+RECGA	1%		3%		5%		7%		10%		15%		25%		37%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
	$\mathcal{S}$																
Number	True positive	0.00	0.10	0.52	0.43	0.55	0.44	0.31	0.50	0.69	0.36	1.01	0.46	1.01	0.38	0.68	0.27
Number	True negative	18.15	17.30	6.57	9.05	7.20	9.12	8.15	7.86	7.08	5.94	3.71	3.61	3.94	3.72	3.61	3.50
Number	True classification	18.15	17.29	6.48	9.03	7.08	9.12	8.13	7.88	7.05	6.07	3.83	3.57	3.98	3.76	3.60	3.46
Ratio	True positive	0.00	0.10	0.13	0.11	0.09	0.07	0.03	0.06	0.05	0.03	0.05	0.02	0.03	0.01	0.01	0.00
Ratio	True negative	0.15	0.15	0.06	0.08	0.06	0.08	0.07	0.07	0.06	0.05	0.03	0.03	0.03	0.03	0.03	0.03
Ratio	True classification	0.15	0.14	0.05	0.07	0.06	0.07	0.06	0.06	0.05	0.05	0.03	0.03	0.03	0.02	0.02	0.02
	Fscore	0.10	0.13	0.08	0.07	0.06	0.06	0.04	0.05	0.04	0.03	0.03	0.02	0.02	0.02	0.02	0.02
	Fmeasure	0.32	0.33	0.22	0.23	0.20	0.21	0.16	0.17	0.13	0.13	0.07	0.06	0.04	0.04	0.02	0.02
	$\mathcal{V}$																
Number	True positive	0.00	0.00	0.48	0.34	0.59	0.52	0.87	0.65	1.02	1.09	1.34	1.40	1.72	1.92	2.08	2.03
Number	True negative	17.55	16.34	6.81	8.97	7.84	8.92	7.08	7.23	6.64	5.65	4.25	4.06	4.19	4.19	3.65	3.37
Number	True classification	17.55	16.34	6.85	8.95	7.85	8.81	7.07	7.11	6.53	5.25	4.16	3.87	4.33	3.73	3.54	2.71
Ratio	True positive	0.00	0.00	0.12	0.08	0.10	0.09	0.10	0.07	0.08	0.08	0.06	0.07	0.04	0.05	0.03	0.03
Ratio	True negative	0.15	0.14	0.06	0.08	0.07	0.07	0.06	0.06	0.06	0.05	0.04	0.03	0.04	0.04	0.03	0.03
Ratio	True classification	0.15	0.14	0.06	0.07	0.06	0.07	0.06	0.06	0.05	0.04	0.03	0.03	0.03	0.02	0.02	0.01
	Fscore	0.10	0.09	0.08	0.06	0.06	0.05	0.06	0.04	0.05	0.04	0.04	0.03	0.03	0.02	0.02	0.01
	Fmeasure	0.32	0.31	0.24	0.24	0.22	0.22	0.14	0.15	0.13	0.11	0.07	0.06	0.04	0.04	0.02	0.02
	$\tilde{\mathcal{S}}$																
Number	True positive	0.45	0.42	0.98	0.94	1.32	1.28	1.25	1.32	1.64	1.65	2.08	2.04	2.07	2.26	2.59	2.29
Number	True negative	17.66	16.87	6.02	9.02	7.25	8.79	7.08	7.09	6.44	5.36	4.49	4.11	4.69	4.78	4.74	4.10
Number	True classification	17.58	16.73	5.79	8.64	6.93	8.46	6.95	6.77	6.17	5.05	4.37	4.13	4.56	4.43	5.04	4.33
Ratio	True positive	0.45	0.42	0.25	0.24	0.22	0.21	0.14	0.15	0.13	0.13	0.10	0.10	0.05	0.06	0.04	0.03
Ratio	True negative	0.15	0.14	0.05	0.08	0.06	0.07	0.06	0.06	0.05	0.05	0.04	0.03	0.04	0.04	0.04	0.03
Ratio	True classification	0.15	0.14	0.05	0.07	0.06	0.07	0.05	0.05	0.05	0.04	0.03	0.03	0.03	0.03	0.03	0.02
	Fscore	0.42	0.39	0.19	0.17	0.16	0.15	0.08	0.09	0.07	0.07	0.05	0.05	0.03	0.03	0.03	0.02
	Fmeasure	0.31	0.28	0.19	0.17	0.20	0.19	0.14	0.14	0.12	0.11	0.08	0.07	0.05	0.04	0.03	0.03
	Time (sec.)																
		22.61	13.44	10.72	34.06	4.23	16.15	5.28	15.75	5.66	15.90	6.95	18.66	10.13	23.40	15.49	26.60

Table L.7: Average Performance Results of Random+RECGA (WBCD Dataset)

Random+RECGA		1%		3%		5%		7%		10%		15%		25%		37%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
$\mathcal{V}$																	
Number	True positive	1.00	1.00	4.00	3.99	5.94	5.93	8.84	8.90	12.88	12.85	20.68	20.65	38.59	38.55	67.12	66.88
	True negative	66.74	57.83	66.84	61.41	67.54	63.08	69.62	65.73	71.21	69.86	77.52	75.09	85.11	83.79	99.26	98.32
	True classification	67.74	58.83	70.84	65.40	73.48	69.01	78.46	74.63	84.09	82.71	98.20	95.74	123.70	122.34	166.38	165.20
Ratio	True positive	1.00	1.00	1.00	1.00	0.99	0.99	0.98	0.99	0.99	0.99	0.98	0.98	0.96	0.96	0.96	0.96
	True negative	0.56	0.49	0.56	0.52	0.57	0.53	0.59	0.55	0.60	0.59	0.65	0.63	0.72	0.70	0.83	0.83
	True classification	0.56	0.49	0.58	0.53	0.59	0.55	0.61	0.58	0.64	0.63	0.70	0.68	0.78	0.77	0.88	0.87
	Fscore	0.72	0.65	0.72	0.68	0.72	0.69	0.73	0.71	0.74	0.73	0.78	0.77	0.82	0.81	0.89	0.89
	Fmeasure	0.04	0.03	0.13	0.12	0.19	0.18	0.26	0.25	0.35	0.35	0.50	0.49	0.69	0.68	0.86	0.85
$\tilde{\mathcal{S}}$																	
Number	True positive	0.94	1.00	3.87	3.98	5.92	5.98	8.80	8.91	12.70	12.72	20.53	20.55	39.10	39.11	66.63	66.53
	True negative	66.64	61.25	68.01	64.40	68.15	65.88	70.44	68.61	72.66	72.56	79.17	78.35	86.63	86.64	99.79	99.77
	True classification	67.58	62.25	71.88	68.38	74.07	71.86	79.24	77.52	85.36	85.28	99.70	98.90	125.73	125.75	166.42	166.30
Ratio	True positive	0.94	1.00	0.97	1.00	0.99	1.00	0.98	0.99	0.98	0.98	0.98	0.98	0.98	0.98	0.95	0.95
	True negative	0.56	0.51	0.57	0.54	0.57	0.55	0.59	0.58	0.61	0.61	0.67	0.66	0.73	0.73	0.84	0.84
	True classification	0.56	0.52	0.58	0.56	0.59	0.57	0.62	0.61	0.65	0.65	0.71	0.71	0.79	0.79	0.88	0.88
	Fscore	0.67	0.68	0.71	0.70	0.72	0.71	0.74	0.73	0.75	0.75	0.79	0.79	0.83	0.83	0.89	0.89
	Fmeasure	0.04	0.03	0.13	0.13	0.19	0.19	0.27	0.26	0.35	0.35	0.51	0.50	0.70	0.70	0.86	0.85
Time (sec.)																	
		50.76	16.58	52.06	17.68	52.55	18.72	54.81	18.67	56.06	19.81	62.51	22.25	79.98	27.85	113.11	39.31

Table L.8: Standard Deviations of Performance Indicators of Random+RECGA (WBCD Dataset)

Random+RECGA		1%		3%		5%		7%		10%		15%		25%		37%	
		H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
$\mathcal{V}$																	
Number	True positive	0.00	0.00	0.10	0.00	0.29	0.24	0.33	0.44	0.36	0.32	0.54	0.53	0.94	0.97	1.47	1.54
	True negative	7.26	6.32	6.87	6.21	6.92	5.52	6.74	5.67	6.52	6.33	6.19	5.88	5.38	4.62	3.64	4.05
	True classification	7.26	6.32	6.87	6.21	6.95	5.52	6.65	5.54	6.52	6.36	6.29	6.00	5.19	4.43	3.87	4.41
Ratio	True positive	0.00	0.00	0.02	0.00	0.05	0.04	0.04	0.05	0.03	0.02	0.03	0.03	0.02	0.02	0.02	0.02
	True negative	0.06	0.05	0.06	0.05	0.06	0.05	0.06	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.03	0.03
	True classification	0.06	0.05	0.06	0.05	0.06	0.04	0.05	0.04	0.05	0.05	0.04	0.04	0.03	0.03	0.02	0.02
	Fscore	0.06	0.04	0.05	0.04	0.05	0.04	0.05	0.04	0.04	0.04	0.04	0.04	0.03	0.03	0.02	0.02
	Fmeasure	0.00	0.00	0.01	0.02	0.02	0.02	0.02	0.02	0.03	0.03	0.04	0.04	0.03	0.03	0.02	0.03
$\widetilde{\mathcal{S}}$																	
Number	True positive	0.00	0.24	0.14	0.39	0.14	0.31	0.32	0.53	0.53	0.54	0.62	0.66	0.88	0.87	1.47	1.45
	True negative	6.63	7.74	6.42	7.20	6.58	6.92	6.32	6.54	5.99	6.12	5.38	5.67	4.59	4.39	3.43	3.68
	True classification	6.63	7.73	6.44	7.14	6.60	6.94	6.30	6.51	5.88	6.02	5.33	5.71	4.48	4.30	3.45	3.69
Ratio	True positive	0.00	0.24	0.04	0.10	0.02	0.05	0.04	0.06	0.04	0.04	0.03	0.03	0.02	0.02	0.02	0.02
	True negative	0.06	0.07	0.05	0.06	0.06	0.06	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.04	0.03	0.03
	True classification	0.06	0.06	0.05	0.06	0.05	0.06	0.05	0.05	0.04	0.05	0.04	0.04	0.03	0.03	0.02	0.02
	Fscore	0.05	0.18	0.05	0.05	0.05	0.05	0.04	0.04	0.04	0.04	0.03	0.04	0.02	0.02	0.02	0.02
	Fmeasure	0.00	0.01	0.01	0.02	0.02	0.02	0.03	0.03	0.03	0.03	0.03	0.04	0.03	0.03	0.02	0.02
Time (sec.)																	
		0.41	8.40	1.07	8.01	1.73	8.29	0.40	7.91	0.46	6.06	0.45	7.57	0.43	6.79	0.55	10.69

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name	: Şahin Mahmutoğulları, Halenur
Nationality	: Turkish (TC)
Date and Place of Birth	: 11 January 1990, Erzurum
Phone	: +90 536 877 44 57
E-mail	: halenur.sahin@metu.edu.tr

EDUCATION

Degree	**Institution**	**Year of Graduation**
Visiting scholar	Georgia Institute of Technology	October 2017- July 2018
MS	Bilkent University Industrial Engineering	2013
BS	Bilkent University Industrial Engineering	2011

FOREIGN LANGUAGES

Turkish (Native), English (Advanced), French (Basic), Japanese (Basic), Russian (Basic), Latin (Basic)

PUBLICATIONS

1. Sahin H., Kara B. Y., Karasan O. E., Debris removal during disaster response: A case for Turkey. Socio-Economic Planning Sciences, 53 (2016): 49-59.

2. Sahin H., Duran S., Yakici E., Sahin M., Patient classication considering the risk of restenosis after coronary stent placement. Journal of Heuristics 25.4-5 (2019): 703-729.

3. Sahin H., Duran S., Yakici E., A rare event classification model for binary response variables. (In progress).

4. Sahin H., Duran S., Yakici E., Sahin M., Can we trust machine learning methods for diagnosis of coronary in-stent-restenosis (In progress).

5. Sahin H., Keskinocak P., Paynabar K., Mortality due to nicotine dependence. (In progress)