AUC MAXIMIZATION FOR BINARY CLASSIFICATION USING COMBINATORIAL BENDERS CUTS

A Thesis

by

İbrahim Edhem Sakarya

Submitted to the
Graduate School of Sciences and Engineering
In Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Industrial Engineering

Özyeğin University
June 2019

AUC MAXIMIZATION FOR BINARY CLASSIFICATION USING COMBINATORIAL BENDERS CUTS

Approved by:

Professor Ö. Erhun Kundakcıoğlu, Advisor
Department of Industrial Engineering
Özyeğin University

Professor Okan Örsan Özener
Department of Industrial Engineering
Özyeğin University

Professor Tonguç Ünlüyurt
Faculty of Engineering and Natural Sciences
Sabancı University

Date Approved: 11 June 2019

To my family, to whom I owe my strength

For their advice, their patience, and their faith

ABSTRACT

The purpose of this study is to maximize the area under Receiver Operating Characteristic curve for binary classification problems using a scoring-based mixed integer linear programming formulation. We investigate exact approaches using a reformulation, combinatorial Benders cuts, and heuristic bounding methods. Our study presents computational results on benchmark datasets and paves the way for future studies on scoring-based approaches.

ÖZETÇE

Bu çalışmanın amacı ikili sınıflandırma problemleri için Alıcı İşletim Karakteristiği (ROC) eğrisi altındaki alanı (AUC) puanlamaya dayalı bir karışık tamsayı izlenceleme gösterimi kullanarak doğrudan maksimize etmektir. Çalışmamızda pekin yöntemleri; yeniden gösterimleri, sezgisel sınırlama yöntemlerini ve birleşi Benders kesilerini kullanarak inceledik. Çalışmamız, denektaşı veri setleri üzerindeki sayısal hesaplama sonuçlarını sunar ve puanlamaya dayalı yaklaşımlarla ilgili gelecek çalışmaların yolunu açar.

ACKNOWLEDGEMENTS

I would like to express my very great appreciation to my advisor, Dr. Ömer Erhun Kundakcıoğlu for guiding and supporting me during my whole graduate school life. He has set an example of excellence as a researcher, mentor, instructor, and role model.

I would like to thank all professors in the Department of Industrial Engineering, especially Dr. Erinç Albey, who was my advisor in the senior year of my bachelor study.

My sincere thanks also goes to members of our research group, and especially to my fellow friends Milad Elyasi, Tonguç Yavuz, and Yurtsev Mıhçıoğlu.

Finally, I must express my very profound gratitude to my parents, my sister, and to my little brother for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Supervised learning is one of the two fundamental paradigms of machine learning (ML). In contrast to unsupervised learning, which is another type of ML, labels for data instances are known in supervised learning. In supervised learning, data instances are represented by pairs of ($\boldsymbol{x_i}$,y_i) where $\boldsymbol{x_i}$ stands for the set of features and y_i is corresponding label to those features for each data instance. When the labels are continuous, the task is called regression. If the labels take discrete values, the task is called classification. In both types of supervised learning the ultimate goal is to learn a mapping function from $\boldsymbol{x_i}$ to output y_i [1].

Classification is one of the tasks that is most frequently carried out by intelligent systems and a large number of techniques have been developed in order to achieve well-designed classification. Some of the best-known supervised classification techniques are support vector machines (SVMs), decision trees, discriminant analysis and instanced-based learning methods [2]. In supervised learning, classification is divided into binary, multi-class, multi-labelled, and hierarchical tasks. Amongst them, binary classification is the most popular classification type, where the data instances are classified into one, and only one, of two non-overlapping classes [3]. In addition, binary classification is currently being applied in numerous fields such as medical diagnosis, fraud detection, credit risk categorization, and text retrieval in real life [4, 5, 6, 7].

A classifier's correctness can be evaluated by using the number of the correctly classified as positives (True Positives), correctly classified as negatives (True Negatives), incorrectly classified as positives (False Positives) and incorrectly classified as negatives (False Negatives). These four numbers constitute the confusion matrix as shown in Table 1.

Table 1: Confusion matrix for binary classification

Actual Class	**Classified as *Positives***	**Classified as *Negatives***
positive	True Positives (TP)	False Negatives (FN)
negative	FP Positives (FP)	True Negatives (TN)

There are several measures for binary classification based on the confusion matrix. Most often used measures are *Accuracy, Precision, Recall, F-Score, Specificity* and *area under Receiver Operating Characteristic (ROC) Curve (AUC)* [3]. We do not delve into details of calculation methods of these metrics. On the other hand, it is important to emphasize that simple classification accuracy is often a poor metric [8, 9], and among all others, AUC appears to be one of the best ways to measure a classifier's performance [10]. Furthermore, using AUC for selecting classification models provides better accuracy in validation set than using accuracy for selecting models [11].

Algorithms with loss functions and designed for error rate minimization such as neural networks do not necessarily lead to the best AUC possible [12, 13, 14]. However, AUC represents the probability of correctly ranking a randomly chosen pair of positive and negative instances [15]. Therefore, AUC can be maximized, by maximizing the number correctly ranked pairs.

Moreover, maximizing the AUC by pairwise ranking further leads to *maximum feasible subsystem* (MAX FS) problem since satisfying maximum number of constraints is aimed, and a scoring-based mixed integer linear programming formulation as a variation of maximum feasible subsystem problem already exists in literature. This thesis focuses on maximizing area under Receiver Operating Characteristic Curve for binary classification problems by investigating a scoring-based mixed integer optimization model using a reformulation, combinatorial Benders cuts, and heuristic bounding methods.

The remainder of this study is organized as follows: Chapter 2 provides a brief review of the literature related to binary classification and AUC maximization. Chapter 3 presents the problem definition and presented solution approaches. Chapter 4 contains our computational results on benchmark data sets. Chapter 5 concludes the thesis providing directions for future research.

CHAPTER II

LITERATURE REVIEW

2.1 Motivation

Machine Learning has several applications and the most significant one is predictive data mining. The datasets for ML algorithms are built up by data instances and each data instance is represented by a set of features. Features of data instances may be binary, continuous or categorical [16]. If the data instances are given with the corresponding known labels (outputs), the task is called *supervised learning*, if the labels are not know, then it is called *unsupervised learning* [17]. In supervised learning, if the labels of data instances take discrete values and the goal is to splitting up data instances so that each is assigned to one of number mutually exhaustive and exclusive categories known as classes (i.e., they cannot be assigned more than one class), the task is called classification. Classification occurs significantly frequently in our everyday life and many decision-making processes can be formulated as classification problems [1, 18]. Some real life examples of classification are such as categorization of people if they are possible customers of a specific product or not depending on their previous shopping choices and dividing up the credit card applicants into those in who has high-risk, medium-risk and low-risk depending on their salaries and credit scores. There are several types of classification approaches and the binary classification, where data instances are classified in to one of two classes, is the most popular type of classification [3].

2.2 Related Work

We categorize the related work as binary classification algorithms, performance metrics of binary classification, MAX FS problem, and AUC maximization problem.

2.2.1 Binary Classification

There are numerous algorithms to solve binary classification problems such as support vector machines, decision trees, discriminant analysis and instanced-based learning methods. SVMs and decision trees also involves mathematical programming and are stated in this study, the readers who are interested on binary classification methods can find further information in [16].

Decision trees classify data instances based on the sorted values of data instances' attributes. Each node in decision trees represents an attribute-based tests and contains a branch for every possible outcome of the test. Leaves of decision trees represent the corresponding classes that the data instance belongs. Starting from the root node, attribute based tests are evaluated. Algorithm continues until a leaf node is encountered. The feature that best divides the data should be root node of the decision tree and constructing an optimal binary decision tree is NP-complete problem, therefore efficient heuristics and constructing near optimal trees have been a search for many researchers [2, 19]. Some of the well-know algorithms for constructing decision trees are proposed in [20, 21, 22]. In addition, linear programming is utilized for determining linear combination splits within binary decision trees in [23] and for finding optimal multivariate splits at the nodes of decision trees in [24]. However based on the multi-disciplinary survey in [25], there is no single best method for constructing decision trees.

Support Vector Machines classify data instances based on their distance to separating hyperplane and is proposed as a maximum-margin classifier [26, 27]. SVM maximizes the margin on each side of separating hyperplane data instances. If the dataset is linearly separable, then there exists a pair $(\mathbf{w}, b)$ such that

$$\mathbf{w}^T\mathbf{x}_i + b \geq 1 \qquad \forall i \in P \tag{1}$$

$$\mathbf{w}^T\mathbf{x}_i + b \leq -1 \qquad \forall i \in N \tag{2}$$

where $\mathbf{w}$ is normal to the hyperplane, $|b|/||\mathbf{w}||$ is perpendicular distance from origin to hyperplane, and P and N are the set of positive and negative instances respectively. Data points, which lies on the one of the hyperplanes $H_1 : \mathbf{w}^T\mathbf{x_i} + b = 1$ and $H_2 : \mathbf{w}^T\mathbf{x_i} + b = -1$ are called as support vectors and the pair of hyperplanes H_1 and H_2 which gives the maximum margin can be found by minimization of $||\mathbf{w}||^2$ [28, 29]. SVM have wide range of applications, they are used in identification of diseases [30, 31], text categorization [7], object recognition [32] and in many other fields [33]. There are also some studies that combines SVMs with metaheuristic methods such as particle swarm optimization [34].

2.2.2 Performance Metrics

In order to be able to compare classification algorithms, a performance metric for comparison is needed. Therefore, performance metrics have fundamental role on assessing the quality of classification algorithms. There are several performance metrics and classified into three main groups as metrics based on a threshold and a qualitative understanding of error, metrics based on a probabilistic understanding of error and metrics based on how well the model ranks the examples in [13]. An example for each group can be given by accuracy, cross-entropy and area under ROC curve respectively. Performance measures are also classified into two main groups such as metrics represented by scalar values which includes accuracy, sensitivity and specificity and metrics based one graphical assessment methods that contains area under ROC curve and Precision-Recall (PR) curve [35]. Most common used measures for binary classification and their formulas based on confusion matrix in Table 1, are presented in Table 2 [36]. Analysis of these performance metrics in order to decide which one is

Table 2: Binary classification measures

Measure	**Formula**
Accuracy	$\frac{TP+TN}{TP+FN+FP+TN}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
Fscore	$\frac{TP+TN}{TP+FN+FP+TN}$
Specificity	$\frac{(\beta^2\ +\ 1)\times TP}{(\beta^2\ +\ 1)\times TP+\ \beta^2\times FN\ +\ FP}$
AUC	$\frac{1}{2}\left(\frac{TP}{TP\ +\ FN}\ +\frac{TN}{TN\ +\ FP}\ \right)$

best for comparing classification algorithms has been a topic many studies. Classification accuracy is investigated in [9] and shown that it is not a sufficient metric for classifier performance. Superiority of metrics are also investigated when the data is imbalanced [37] and it is proved that the metrics which use values from both columns of confusion matrix, such as accuracy and precision, are significantly sensitive to the imbalanced data as stated in [38]. A deep analysis of AUC is also presented in [8] and shown that the AUC of a classifier is equal to probability of a randomly chosen positive instance to be ranked higher than the randomly chosen negative instance. This shows the equality between Wilcoxon-Mann-Whitney statistic and AUC in discrete cases. Another intense investigation of AUC as a measure of classifier performance is also made in [10]. They utilize two decision trees (C4.5 and Multiscale Classifier); two neural networks (Perceptron and Multi-layered Perceptron); and two statistical methods (K-Nearest Neighbors and a Quadratic Discriminant Function) on six different real world datasets to compare AUC with accuracy and report that AUC should be preferred to accuracy.

2.2.3 MAX FS Problem and AUC Maximization

Since AUC does not confound with imbalanced datasets accuracy or precision, it is a more general and robust measure of classifiers' performance [12]. Therefore there has been many studies to improve the AUC of classifiers. Several approaches are based on error rate minimization, however they do not necessarily optimize the AUC [14]. In [39], the authors propose a characterized SVM that maximizes the AUC and there are also some studies tries to maximize the AUC by metaheuristic methods, such as Simulated Annealing (SA) in [40]. However, non of these methods directly maximizes the AUC and exactly guarantee that the AUC they obtain is the optimal.

AUC is not easy to compute, however, is exactly equal to Wilcoxon-Mann-Whitney (WMW) statistic [14, 15, 41]. Maximization of an approximation of WMW statistic is proposed in [12]. An exact maximization of AUC through WMW statistic by mixed integer programming technique is proposed in [42]. Their formulation tries to satisfy as many inequality as possible from the WMW statistic set, and is a special structured version of maximum feasible subsystem problem where all inequalities are in same shape and the infeasibility of linear set is unknown.

CHAPTER III

PROBLEM DEFINITION AND SOLUTION APPROACHES

3.1 Problem Definition

In this study, we focus on maximizing the area under ROC curve for binary classification problems, by utilizing Wilcoxon-Mann-Whitney statistic that leads to a *maximum feasible subsystem* problem. MAX FS problem is considered to determine a feasible subsystem containing as many inequalities as possible from a given infeasible set of constraints. It has been proved to be an NP-hard problem and also difficult to approximate [43]. Mathematical optimization model of our problem is formulated as a special case of MAX FS problem in [42]. This model cannot be solved to optimality in many data sets in reasonable time, and our aim is to solve the problem to optimality if possible or obtain a smaller optimality gap in a given time limit.

3.1.1 Maximizing the Area Under ROC Curve

The ROC curve was first developed in the 1950s to detect signals [15]. The curve consists of False Positive Rate (FPR) on the x axis and True Positive Rate (TPR) on the y axis. By shifting the threshold from most positive (i.e. classifying all instances as negatives) to most negative (i.e., classifying all instances as positives), the points on the curve are obtained. For a random classification, it is expected to obtain a straight line from (0,0) to (1,1). Therefore, a classifier, which performs better than random classification, should provide an ROC curve which is above this straight line.

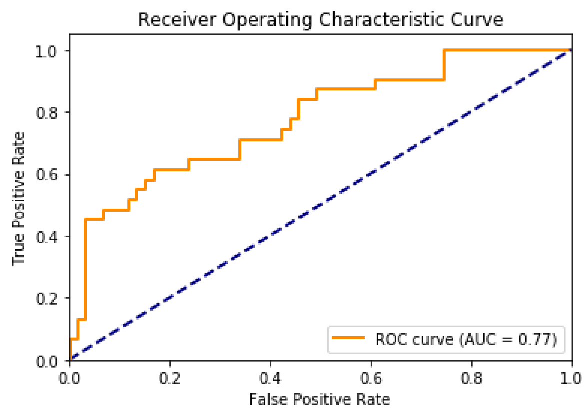

Figure 1: An example of ROC Curve

There is an example of ROC curve in Figure 1 and the AUC is defined as the area under this curve. AUC value is exactly the probability P(X>Y) where X is the random variable corresponding to the distribution of the outputs for the positive examples and Y is the one corresponding to the negative examples [44]. Aforementioned probability, the AUC, is equal to Wilcoxon-Mann-Whitney (WMW) statistic (1) in discrete cases [15].

$$\frac{\sum_{i=1}^{k}\sum_{j=1}^{l} 1_{m_i>n_j}}{kl} \tag{1}$$

Where m_1,...,m_k and n_1,...,n_l are the outputs of a fixed classifier for positive and negative data points respectively. $1_{m_i>n_j}$ denotes a binary indicator that takes value 1 if the score of positive instance (m_i) is grater than the score of negative instance (n_j) and 0 otherwise.

3.1.2 Maximum Feasible Subsystem Problem

MAX FS problem finds a feasible subsystem containing as many inequalities as possible from a given infeasible system [45]. This problem can also be considered as finding the minimum number of constraints to remove, in order to resolve infeasibility [46, 47], which is known as *minimum unsatisfied linear relation problem* (MIN ULR) [48]. An infeasible set of constraints can be feasible by deleting at least one member of every *irreducible infeasible subsystem* (IIS) it contains. One other complementary problem is the *minimum-cardinality IIS set-covering* problem (MIN IIS COVER), in which the smallest set of constraints looked for covering all IISs of infeasible system [49].

Similarly, in our problem, in order to obtain the best AUC possible, a classifier (scoring function) should yield outputs that satisfy the following linear program for the maximum number of (i, j) pairs.

$$m_i > n_j \qquad \forall i \in S_+ \,,\ \forall j \in S_- \tag{2}$$

Where S_+ and S_- are the set of positive and negative data points respectively. If the data set is not linearly separable, then (2) is an infeasible system. This study differs from MAX FS problem inasmuch as the fact that the infeasibility of set 2 is not known beforehand. Our objective is to find a classifier which provides output values that can satisfy the maximum number of constraints in set 2.

3.1.3 Mixed Integer Linear Programming Formulation

MIP model below is formulated in [42] to maximize AUC for the binary classification problems. We will refer to this model as original MIP model in this study. The idea is to find a linear scoring function which yields the highest value through WMW statistic. The parameters in the proposed mathematical model are $\mathbf{x}_i$, $\mathbf{x}_k$ and ϵ. $\mathbf{x}_i$ and $\mathbf{x}_k$ represents the attribute vectors of positive instance i where $i \in S_+$ and negative instance k where $k \in S_-$ respectively and ϵ is a small user-specified constant. The decision variables are as follows:

v_i The score of positive instance i

v_k The score of negative instance k

w_j Coefficients of linear scoring function

Using these decision

$$z_{ik} = \begin{cases} 1, & \text{if instance } i \text{ scores greater than instance } k \\ 0, & \text{otherwise} \end{cases}$$

variables, the problem is modelled as follows:

$$\max \sum_{i \in S_+} \sum_{k \in S_-} z_{ik} \tag{3a}$$

$$\text{subject to } z_{ik} \leq v_i - v_k + 1 - \epsilon \qquad \forall i \in S_+,\ \forall k \in S_- \tag{3b}$$

$$v_i = \mathbf{w}^T \mathbf{x}_i \qquad \forall i \in S_+ \tag{3c}$$

$$v_k = \mathbf{w}^T \mathbf{x}_k \qquad \forall k \in S_- \tag{3d}$$

$$-1 \leq w_j \leq 1 \qquad \forall j \in 1..d \tag{3e}$$

$$z_{ik} \in \{0, 1\} \qquad \forall i \in S_+,\ \forall k \in S_- \tag{3f}$$

Constraint (3b) is the condition of WMW statistic and this assures that the binary decision variable z_{ik} is 1 if $v_i > v_k$, and 0 otherwise. That is, the associated binary variable takes 1 if the positive instance has greater score than the negative instance. Constraints (3c) and (3d) let the scores of corresponding positive and negative instances to be equal to associated output values from the scoring function. Above MIP model, that is directly maximizing AUC, is known to outperform state-of-the-art classification techniques in terms of AUC performance. However, problem cannot be solved to optimality in a reasonable time for many data sets. It means the AUC values obtained through this model could even be further improved in terms of time or best solution found. This is our main motivation to solve the problem to optimality if possible, if not, accelerate the optimization process and obtain smaller optimality gap within a specified time limit.

3.2 Solution Approaches

As mentioned before, original MIP model cannot be solved to optimality for many datasets. We tackle this problem with three different methods, two heuristic methods and one exact method with it's variations. In the first heuristic method, our only intention is to obtain a fast initial feasible solution, which can be used later on. In our second heuristic method, we utilize reformulation techniques to come by with a better initial solution and also to use it iteratively in our third method. In our last method and it's variations, we aim to solve the problem to optimality if we can or reduce the optimality gap within a specified time limit.

3.2.1 Heuristic 1, Based on Linear Relaxation

Original MIP model cannot be solved optimality due to the large number of binary variables. Therefore, we start by solving the associated linear relaxation (LP) of the corresponding MIP problem. We allow binary variables (z_{ik}) to take continuous values in [0,1]. When a positive instance, say i, cannot score greater than a negative instance, say k, z_{ik} can still take fractional values, even though Wilcoxon-Mann-Whitney statistic provides 0 for that z_{ik}. In this case, the objective function gets a higher value than it can feasibly attain. Therefore, the objective function value of this relaxation is not necessarily the AUC but a very optimistic approximation and an upper bound.

Then, as a second step of this method, in the original MIP problem, we set the coefficients of the scoring function, $\boldsymbol{w}$, to the values obtained in linear relaxation form. By doing this we get a feasible solution. As $\boldsymbol{w}$ was obtained in the linear relaxation form, it does not provide an optimal solution, but a lower bound, for the original MIP problem.

3.2.2 Heuristic 2, Based on the Reformulation Linearization Technique

Remember that a binary variable z_{ik} corresponding to a pair of positive-negative instances (i, k), such that $v_i < v_k$, can take fractional value. This is the reason why solving the linear relaxation of original MIP model does not provide AUC exactly. We can solve the linear relaxation of the original problem in a very short amount of time. However we need to tighten the bounds of this linear relaxation to avoid some of the fractional values for z_{ik}. In the first step of this method, we employ a reformulation trick on the original MIP model in order to force the z_{ik}s to take binary values when solving the linear relaxation. Consider the constraint (3b).

$$z_{ik} \leq v_i - v_k + 1 - \epsilon \qquad \forall i \in S_+ \ , \ \forall k \in S_- \tag{4}$$

We first write it in a more compact form

$$z_{ik} \leq \mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_k) + 1 - \epsilon \qquad \forall i \in S_+ \ , \ \forall k \in S_- \tag{5}$$

then we multiply the both sides of inequality by z_{ik} and obtain

$$z_{ik} \times z_{ik} \leq z_{ik} \times (\mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_k) + 1 - \epsilon) \qquad \forall i \in S_+ \ , \ \forall k \in S_- \tag{6}$$

We know that the square of any binary variable is equals either to 1 if the variable take the value of 1 or to 0 if variable take the value of 0. Therefore, the square of a binary variable always equals to itself. Using this well-known property of binary variables, we rewrite the left hand side (LHS) of the inequality as only z_{ik} instead of $z_{ik} \times z_{ik}$.

$$z_{ik} \leq z_{ik} \times (\mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_k) + 1 - \epsilon) \qquad \forall i \in S_+ \ , \ \forall k \in S_- \tag{7}$$

Then using the distributive property, we have

$$z_{ik} \leq z_{ik} \times \mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_k) + z_{ik} - z_{ik} \times \epsilon \qquad \forall i \in S_+ \ , \ \forall k \in S_- \tag{8}$$

after some algebra, we have the following inequality.

$$z_{ik} \times \epsilon \leq z_{ik} \times \mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_k) \qquad \forall i \in S_+ \ , \ \forall k \in S_- \tag{9}$$

At this point, there is still nonlinearity at the right hand side (RHS) of the inequality (9). We reformulate the constraint by substituting the bilinear term $z_{ik} \times \mathbf{w}$ with $\boldsymbol{\eta}_{ik}$ and obtain

$$\epsilon \times z_{ik} \leq \boldsymbol{\eta}_{ik}^T(\mathbf{x}_i - \mathbf{x}_k) \qquad \forall i \in S_+ \ , \ \forall k \in S_- \tag{10}$$

We know that both $\mathbf{w}$ and z_{ik} have upper and lower bounds such that, $\underline{\mathbf{w}_j} \leq \mathbf{w}_j \leq \overline{\mathbf{w}_j}$ and $\underline{z_{ik}} \leq z_{ik} \leq \overline{z_{ik}}$. To make above reformulation valid, we make use of McCormick inequalities [50] and add following constraints to problem.

$$\boldsymbol{\eta}_{ikj} \geq \underline{z_{ik}} \times \mathbf{w}_j + z_{ik} \times \underline{\mathbf{w}_j} - \underline{z_{ik}} \times \underline{\mathbf{w}_j} \qquad \forall i \in S_+ \forall k \in S_- \forall j \in 1...d \tag{11a}$$

$$\boldsymbol{\eta}_{ikj} \geq \overline{z_{ik}} \times \mathbf{w}_j + z_{ik} \times \overline{\mathbf{w}_j} - \overline{z_{ik}} \times \overline{\mathbf{w}_j} \qquad \forall i \in S_+ \forall k \in S_- \forall j \in 1...d \tag{11b}$$

$$\boldsymbol{\eta}_{ikj} \leq \overline{z_{ik}} \times \mathbf{w}_j + z_{ik} \times \underline{\mathbf{w}_j} - \overline{z_{ik}} \times \underline{\mathbf{w}_j} \qquad \forall i \in S_+ \forall k \in S_- \forall j \in 1...d \tag{11c}$$

$$\boldsymbol{\eta}_{ikj} \leq z_{ik} \times \overline{\mathbf{w}_j} + \underline{z_{ik}} \times \mathbf{w}_j - \underline{z_{ik}} \times \overline{\mathbf{w}_j} \qquad \forall i \in S_+ \forall k \in S_- \forall j \in 1...d \tag{11d}$$

Then the final form of the reformulated problem with all newly added constraints is as follows:

$$\max \sum_{i \in S_+} \sum_{k \in S_-} z_{ik} \tag{12a}$$

$$\text{subject to } z_{ik} \leq \mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_k) + 1 - \epsilon \qquad \forall i \in S_+ \text{ , } \forall k \in S_- \tag{12b}$$

$$\underline{\mathbf{w}_j} \leq \mathbf{w}_j \leq \overline{\mathbf{w}_j} \qquad \forall j \in 1...d \tag{12c}$$

$$\underline{z_{ik}} \leq z_{ik} \leq \overline{z_{ik}} \qquad \forall i \in S_+ \forall k \in S_- \tag{12d}$$

$$\epsilon \times z_{ik} \leq \boldsymbol{\eta}_{ik}^T(\mathbf{x}_i - \mathbf{x}_k) \qquad \forall i \in S_+ \forall k \in S_- \tag{12e}$$

$$\boldsymbol{\eta}_{ikj} \geq \underline{z_{ik}} \times \mathbf{w}_j + z_{ik} \times \underline{\mathbf{w}_j} - \underline{z_{ik}} \times \underline{\mathbf{w}_j} \qquad \forall i \in S_+ \forall k \in S_- \forall j \in 1...d \tag{12f}$$

$$\boldsymbol{\eta}_{ikj} \geq \overline{z_{ik}} \times \mathbf{w}_j + z_{ik} \times \overline{\mathbf{w}_j} - \overline{z_{ik}} \times \overline{\mathbf{w}_j} \qquad \forall i \in S_+ \forall k \in S_- \forall j \in 1...d \tag{12g}$$

$$\boldsymbol{\eta}_{ikj} \leq \overline{z_{ik}} \times \mathbf{w}_j + z_{ik} \times \underline{\mathbf{w}_j} - \overline{z_{ik}} \times \underline{\mathbf{w}_j} \qquad \forall i \in S_+ \forall k \in S_- \forall j \in 1...d \tag{12h}$$

$$\boldsymbol{\eta}_{ikj} \leq z_{ik} \times \overline{\mathbf{w}_j} + \underline{z_{ik}} \times \mathbf{w}_j - \underline{z_{ik}} \times \overline{\mathbf{w}_j} \qquad \forall i \in S_+ \forall k \in S_- \forall j \in 1...d \tag{12i}$$

$$-1 \leq \mathbf{w}_j \leq 1 \qquad \forall j \in 1...d \tag{12j}$$

$$z_{ik} \in \{0, 1\} \qquad \forall i \in S_+ \forall k \in S_- \tag{12k}$$

At this point, when we solve the linear relaxation of reformulated model (LP-RLT) above, we have only two outcomes for z_{ik}. Consider that, the RHS of the inequality (12b), is greater than or equal to 1 (i.e., $\mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_k) + 1 - \epsilon \geq 1$) then z_{ik} can get a value of 1. If RHS of the inequality is less than 1, then thanks to our reformulation (12e), any of the z_{ik}s cannot get fractional value but a value of 0. However, there is still a potential issue caused by linearization. Linearization of product of two continuous variables with McCormick inequalities does not yield an exact linearization. They are shown to be envelopes in [51]. That is why we cannot force all of the z_{ik}s to not take fractional values but rather some of them. Therefore, the optimal solution of this problem is not equal to AUC.

In the second step of this method, to obtain a feasible solution, reflecting exactly the AUC, we use the same approach with Heuristic 1 in 3.2.1. We set the coefficients of scoring function, $\boldsymbol{w}$, to the ones obtained in linear relaxation of reformulated original MIP model. Even though there are still fractional values on the optimal solution of linear relaxation of reformulated original MIP model, the number of fractional values is significantly less than the number obtained in the first step of Heuristic 1. Since it has a tighter formulation than Heuristic 1, the solution of this problem provide a better (lower) upper bound for the original problem; hence, it is clearly more realistic than the Heuristic 1.

3.2.3 An Exact Approach Based on Benders Decomposition

Benders decomposition is an exact solution method to solve large-scale optimization problems. Instead of considering all of the decision variables and constraints of a large-scale problem at the same time, Benders decomposition divides problem into multiple relatively easily solvable problems (Master problem and subproblem(s)) [52]. Original MIP model has only one set of constraints which contains both binary and continuous variables, while solving the problem with classical Benders decomposition, we might have to visit all corner points. Therefore, application of the classical Benders decomposition does not work effectively for our problem. In order to tackle this issue, we modify the classical Benders decomposition with respect to our problem's structure.

3.2.3.1 Classical Benders Decomposition

Integer variables are generally considered to be complicating variables. In a sense that if these variables are fixed to some specific values, remaining part of the problem is an easy to solve linear problem with non-complicating variables. Consider the optimization problem 13, where $\mathbf{x}$ is a set of non-negative continuous variables and $\mathbf{y}$ is a set of integer variables.

$$\max \quad \mathbf{c}^T\mathbf{x} + \mathbf{f}^T\mathbf{y} \tag{13a}$$

$$\text{subject to} \quad A\mathbf{x} + B\mathbf{y} \leq \mathbf{b} \tag{13b}$$

$$\mathbf{x} \geq 0 \tag{13c}$$

$$\mathbf{y} \in Y \tag{13d}$$

The algorithm is initialized with a lower bound (LB) and an upper bound (UB) equal to $-\infty$ and $+\infty$ respectively. Initially master problem (MP) is maximization of a newly introduced variable, say z, subject to the constraints where integer variables restricted to be in the same domain as original problem only. Then the MP is,

$$\max \quad z \tag{14a}$$

$$\text{subject to} \quad LB \leq z \leq UB \tag{14b}$$

$$\mathbf{y} \in Y \tag{14c}$$

When $\mathbf{y}$ is fixed to some $\mathbf{y}^*$ with respect to the constraint (14c), the subproblem (SP), which contains only continuous variables, is obtained as following:

$$\max \quad \mathbf{c}^T\mathbf{x} \tag{15a}$$

$$\text{subject to} \quad A\mathbf{x} \leq \mathbf{b} - B\mathbf{y}^* \tag{15b}$$

$$\mathbf{x} \geq 0 \tag{15c}$$

$$\tag{15d}$$

In the following steps of Benders decomposition, the dual of SP is solved. If the dual is unbounded, then the primal (SP) is infeasible, therefore the original problem 13 is infeasible for such $\mathbf{y}^*$. In this case, by using the extreme ray ($\mathbf{u}^*$) of the dual of SP, a feasibility cut (16) is added to MP.

$$[\mathbf{b} - B\mathbf{y}]^T\mathbf{u}^* \geq 0 \tag{16}$$

If the dual of SP is solved to optimality, according to strong duality, SP has the same objective function value. In this case, LB is updated as follows:

$$LB = \max\{LB, \mathbf{f}^T\mathbf{y}^* + [\mathbf{b} - B\mathbf{y}^*]^T\mathbf{u}^*\} \tag{17}$$

When $\mathbf{u}^*$ is acquired, the objective function value of problem 13 can be written as a function of $\mathbf{y}$. Then an optimality cut (18) is added to the MP.

$$z \leq \mathbf{f}^T\mathbf{y} + [\mathbf{b} - B\mathbf{y}]^T\mathbf{u}^* \tag{18}$$

After both cases, the MP is solved with added cuts, $\mathbf{y}^*$ is set to $\mathbf{y}$ values in MP's solution, and UB is updated, such as $UB = z^*$. These steps are followed until a given convergence criteria (e.g: $UB - LB \leq \epsilon$) is met.

3.2.3.2 Benders Decomposition with Modified Combinatorial Cuts

While solving the original MIP model, algorithm takes too long to converge if the classical steps of Benders decomposition are followed. Therefore we modify the algorithm. We first choose our LB and UB more wisely. Instead of setting them to $-\infty$ and $+\infty$, we set UB to the optimal objective function value of the linear relaxation of reformulated model (12) and LB to the objective function value of Heuristic 2. Then we initialize the algorithm by directly solving MP (19) without any cuts from subproblems and obtain z_{ik}^*.

$$\begin{aligned}
\max \quad & Q && && \text{(19a)}\\
\text{subject to} \quad & \sum_{i \in S_+} \sum_{k \in S_-} z_{ik} \leq UB && \forall i \in S_+ \;,\; \forall k \in S_- && \text{(19b)}\\
& \sum_{i \in S_+} \sum_{k \in S_-} z_{ik} \geq LB && \forall i \in S_+ \;,\; \forall k \in S_- && \text{(19c)}\\
& \sum_{i \in S_+} \sum_{k \in S_-} z_{ik} \geq Q && \forall i \in S_+ \;,\; \forall k \in S_- && \text{(19d)}\\
& z_{ik} \in \{0,1\} && \forall i \in S_+ \;,\; \forall k \in S_- && \text{(19e)}\\
& Q \geq 0 && && \text{(19f)}
\end{aligned}$$

Then, in each iteration, in contrast to classical Benders decomposition, we do not solve MP to optimality and add cuts by solving subproblem, instead, we never stop solving the MP, only interrupt it by using callbacks and adding lazy constraints. While solving the MP, whenever Gurobi finds a feasible integer solution (MIP Node), our algorithm invokes a predefined callback function. In our callback function, we first set up and solve the SP (20) with given z^*_{ik} values.

$$\max \ 0 \tag{20a}$$

$$\text{subject to} \quad z^*_{ik} \leq \mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_k) + 1 - \epsilon \quad (\mathbf{u}^*) \qquad \forall i \in S_+ \ , \ \forall k \in S_- \tag{20b}$$

$$-1 \leq \mathbf{w}_j \leq 1 \qquad \forall j \in 1..d \tag{20c}$$

Due to the structure of original MIP model, where there is no continuous variables in the objective function, our SP does not have an objective function. Basically we seek a feasibility in the SP. Considering the objective function of MP, MP would always set maximum number z_{ik}'s to 1. Therefore, in contrast to classical Benders decomposition, in any iteration, if the z^*_{ik} set is feasible for the SP, then it is guaranteed that same set of z^*_{ik} is also optimal for the original problem. If the given z^*_{ik} from MP is not feasible for the SP, then we the dual of SP has extreme rays. This is the case in each iteration until an optimal solution for original problem found. For this reason, in each iteration, we add a classical feasibility cut, in the shape of (16), to MP by using the associated dual variables ($\mathbf{u}^*$) of SP. However we do not continue to solve the MP just after feasibility cut is added. Adding only this cut is not enough for problem to converge in a reasonable time. Therefore, we continue to solve some other subproblems, in order to generate better cuts for MP. We first solve the same SP for a subset (S) of data instances. For example, we randomly choose p positive instances and n negative instances. Solve the SP only for fixed z^*_{ik}, taken from the solution of MP, where $i \in P, \ k \in N, \ |P| = p, \ |N| = n, \ P \subseteq S$ and $N \subseteq S$. If SP is feasible for S, then we continue without adding cut. If it is infeasible, we add a classical feasibility cut first. Then, we solve original MIP model only for this subset. As we select relatively small numbers for p and n, this restricted original MIP model (ROP) is solved in seconds. ROP is solved without considering any other instances but S, hence, in any solution, summation of z_{ik} where $i, k \in S$, cannot be greater than the objective function value of ROP (o-ROP). Thus, the cut generated by solving the ROP for S (Subset Cut) is as follows;

$$\sum_{i \in P} \sum_{k \in N} z_{ik} \leq o - ROP \tag{21}$$

We add this subset cut not only to MP but also to LP-RLT. We solve the LP-RLT with added cuts and corresponding Heuristic 2 then update the UB and LB according to their objective function values respectively. We follow these steps until a given convergence criteria is met or a prespecified time limit is reached. Fig. 2 shows a flowchart representing our algorithm.

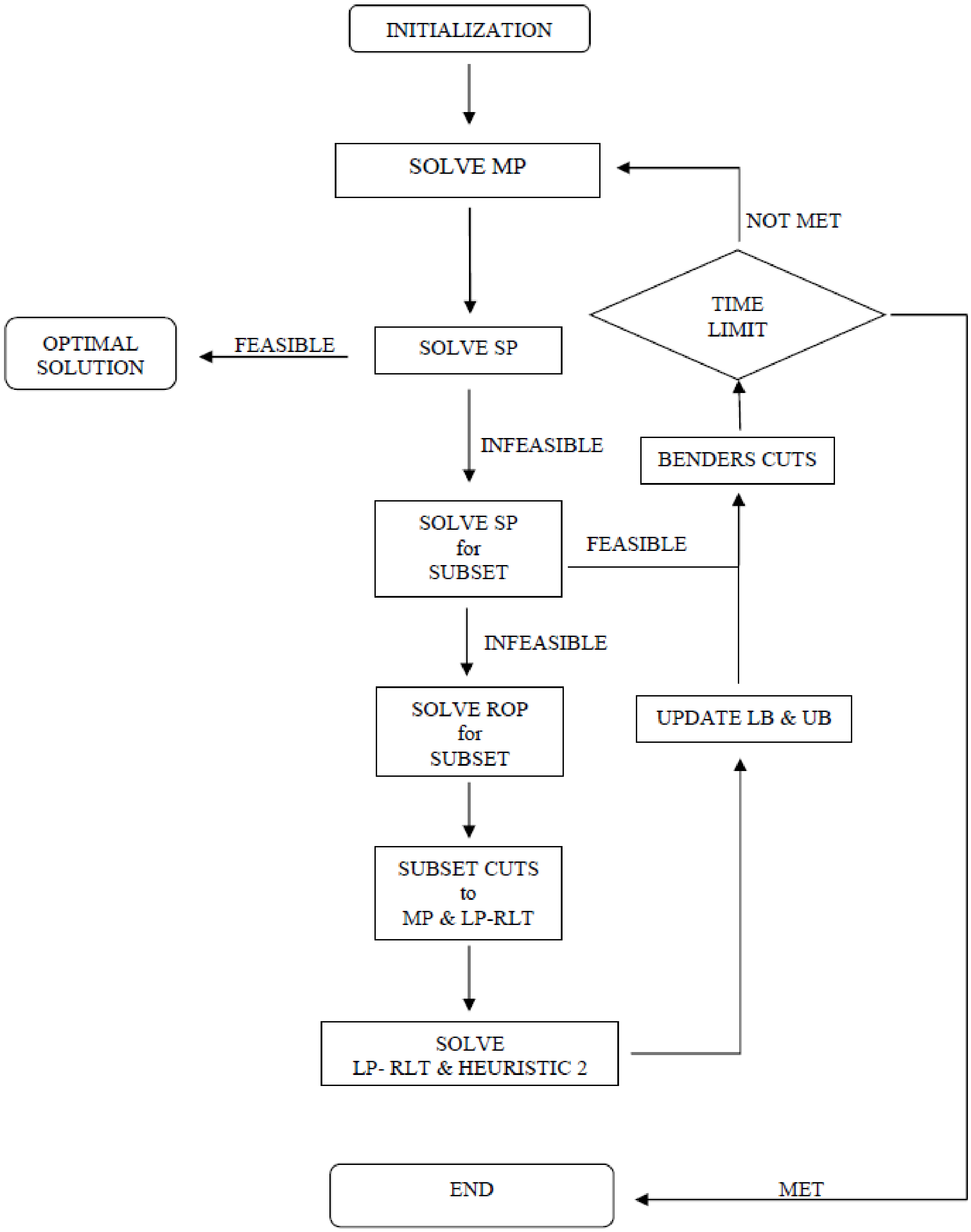

Figure 2: Modified Benders Decomposition Algorithm (MBD)

In each iteration we randomly choose positive and negative instances to generate subsets. We further improve this procedure by choosing the instances wisely. For a given scoring function $\mathbf{w}$, taken from the solution of LP-RLT in previous iteration, we sort all instances according to their scores. We take p_1 of lowest scored positive instances and n_1 of highest scored negative instances. We know that z_{ik} where $v_i \leq v_k$ is going to be 0 in the optimal solution. For this reason a cut, generated by considering these type of (i, k) pairs, would be a better cut than the one generated by randomly chosen pairs. However, if we solve ROP with only low scored positive and high scored negative instances, model would overfit to this subset and provide an unrealistic scoring function ($\mathbf{w}$). To eliminate overfitting, we also add p_2 high scored positive and n_2 low scored negative instances, such as $p_1 + p_2 = p$ and $n_1 + n_2 = n$. Consequently, we select which instance to include in subsets in a more promising way and we will refer to MBDs with wise selection as wise modified Bender decomposition (w-MBD) in this study.

CHAPTER IV

COMPUTATIONAL RESULTS

In this chapter, we present our computational results and compare the performance of our methods against original MIP model on benchmark datasets. It is already known that original MIP model outperform the state-of-the-art classification techniques in terms of AUC performance [42]; thus, we do not split the datasets into training and test sets and delve into cross-validation. Since we do not split the datasets, we normalize each of them as a whole in preprocessing using min-max scaling. We solve the original MIP model, Heuristic 1, Heuristic 2, and modified Benders decomposition methods using 7 datasets. FourClass is from LIBSVM Collection [53] and all others are from UCI Machine Learning Repository. The number of instances in some of the datasets varies due to deleted duplicate rows as suggested by the provider of dataset [54, 55, 56, 57]. The description of datasets is given in Table 3. MBD is solved for 8 different subset sizes depending on the number of positive and negative instances. 1 of these subsets is also solved with the w-MBD in order to investigate the contribution of wise selection method. This experiment is conducted using $\epsilon = 10^{-4}$ to solve all LPs and $\epsilon = 10^{-6}$ to solve all MIPs and all computations are performed using Python, calling Gurobi 8.0 to solve optimization problems, on a 3.5 GHz Intel Xeon (E5-1650 v2) computer with 16 GB DDR3 ECC (1866 MHz) RAM and the macOS HighSierra operating system.

Table 3: Description of datasets

Dataset	Number of Attributes	Number of Instances	Number of Positives	Number of Negatives
Banknote Authentication	5	1372	610	762
Blood Transfusion Service Center	5	748	178	570
Caeserian Section	5	80	46	34
Cryotherapy	7	90	48	42
FourClass	2	862	307	555
Liver Disorders	7	341	142	199
Vertebral Column	6	310	210	100

Since our aim is to maximize area under ROC curve, we choose AUC as a performance metric while comparing the solution approaches including mixed integer programming, heuristic algorithms and Benders decomposition. We also compare the methods based on the amount of time it takes reach similar level of AUC values.

We first compare the performance of our heuristic methods with original MIP model. Table 4 shows the AUC values on each dataset for original MIP model, Heuristic 1 and Heuristic 2; bold indicates the value is the highest AUC on corresponding dataset. Each dataset is solved with a time limit of 1 hour. In each dataset, our second heuristic finds better or almost same AUC value in a very short amount of time.

Table 4: AUC (%) value (top) and runtime (seconds) of methods (bottom)

Dataset	Original MIP	Heuristic 1	Heuristic 2
Banknote Authentication	0.9996	0.9995	**0.9998**
	3600.5986	**8.0161**	1795.2400
Blood Transfusion Service Center	0.7500	0.4151	**0.759**
	3600.7774	**4.6879**	263.5333
Caeserian Section	**0.7615**	0.4092	0.7519
	3600.0172	**0.0717**	0.3499
Cryotherapy	**0.9653**	0.9067	0.9623
	3600.0180	1.1467	**0.5416**
FourClass	0.8320	0.6180	**0.8333**
	3600.0686	**5.0060**	41.4492
Liver Disorders	0.6563	0.3873	**0.7506**
	3600.9770	**1.3812**	15.3527
Vertebral Column	0.9400	0.8568	**0.9410**
	3600.0839	**4.0622**	8.2558

Heuristics are fast, however they do not carry any information about the upper bound of the problem, thus, we do not have information on proximity to optimality. In Table 5 AUC values obtained by modified Benders decomposition with different 8 different subset sizes. MBD with a (p, n) pair in each column represents the selected number of positive and negative instances respectively, in each iteration of MBD for subset cuts. Due to 1 hour time limit in each method, we do not state runtimes repeatedly.

Table 5: AUC (%) value for each method in 1 hour.

Dataset	**MIP**	**MBD (5-10)**	**MBD (8-15)**	**MBD (10-5)**	**MBD (10-10)**	**MBD (13-7)**	**MBD (15-8)**	**MBD (12-12)**	**MBD (12-8)**
Banknote Authentication	0.9996	**0.9998**	**0.9998**	**0.9998**	**0.9998**	**0.9998**	**0.9998**	**0.9998**	**0.9998**
Blood Transfusion Service Center	0.750	0.759	**0.7593**	0.7592	0.7590	**0.7593**	**0.7593**	**0.7593**	0.7592
Caeserian Section	0.7615	0.7551	**0.7839**	0.7596	0.7551	0.7570	0.7577	0.7583	0.7532
Cryotherapy	**0.9653**	0.9638	0.9628	0.9638	0.9633	0.9633	0.9633	0.9639	0.9633
FourClass	0.8320	0.8335	**0.8335**	0.8335	0.8334	0.8335	0.8335	0.8334	0.8335
Liver Disorders	0.6563	0.7501	**0.7511**	0.7507	0.7508	0.7508	0.7505	0.7506	0.7510
Vertebral Column	0.9400	0.9413	**0.9414**	0.9413	0.9411	0.9412	0.9413	**0.9414**	0.9413

There is always a modified Bender decomposition method that performs better than original MIP model in terms of obtained AUC values. Overall, MBD (8-15) performs better than all other MBDs and MBD (12,12) is the second best. With respect to this outcome, we can say that, the higher the subset size, the better AUC values on MBD methods.

As the objective function value of an exact algorithm in a maximization problem is always a lower bound for the optimum objective function value, a method which provides higher AUC value may still have greater optimality gap than others due to larger upper bound. Therefore, we initially compare the AUC values and the optimality gaps of utilized methods separately then investigate the relation between AUC values and optimality gaps. Table 6 shows the optimality gaps corresponding to each method after 1 hour of computation.

Table 6: Optimality Gap (%) in 1 hour.

Dataset	**MIP**	**MBD (5-10)**	**MBD (8-15)**	**MBD (10-5)**	**MBD (10-10)**	**MBD (13-7)**	**MBD (15-8)**	**MBD (12-12)**	**MBD (12-8)**
Banknote Authentication	**0.0002**	0.0002	0.0002	0.0002	0.0002	0.0002	0.0002	0.0002	0.0002
Blood Transfusion Service Center	0.3144	0.3072	**0.3031**	0.3070	0.3044	0.3051	0.3043	0.3052	0.3054
Caeserian Section	**0.0302**	0.1800	0.0636	0.1679	0.1194	0.1272	0.1129	0.0978	0.1121
Cryotherapy	0.0236	0.0243	0.0155	0.0254	0.0175	0.0217	0.0165	**0.0134**	0.0149
FourClass	0.1994	0.1985	0.1965	0.1985	0.1987	0.1971	0.1969	**0.1959**	0.1972
Liver Disorders	0.5230	0.3235	**0.3141**	0.3236	0.3146	0.3168	0.3151	0.3155	0.3155
Vertebral Column	0.0584	0.0599	0.0568	0.0601	0.0571	0.0576	0.0565	**0.0539**	0.0576

Considering the Table 5 and Table 6, even though original MIP model can provide better AUC value, it might still has larger optimality gap with respect to MBD methods. Surprisingly, in some datasets, original MIP model provides smaller optimality gap even though its AUC is less than the MBD methods. Consider MBD(8-15) in Caeserian Section and Cryotherapy datasets, it provides a clear example for stated situation. The reason behind this outcome can be explained by the effectiveness of subset cuts on MP. They might provide cuts that make MP to have smaller upper bound than original MIP model when MP does not have better lower bound than the original MIP, which results in better optimality gap and worse AUC for modified Benders decomposition. On the other hand, subset cuts may not work effectively when the lower bound of MP is greater than original MIP model and consequently, higher lower bound and greater optimality gap for MP.

We solve the w-MBD only for subset pair (12-12), therefore we compare its results only with MBD (12-12) for each dataset. Table 7 shows the AUC performances and optimality gaps of w-MBD (12-12) and MBD (12-12) for each dataset.

Table 7: AUC (%) values (top) and optimality gaps (bottom) of MBD (12-12) and w-MBD (12-12) methods in 1 hour.

Dataset	**MBD (12-12)**	**w-MBD (12-12)**
Banknote Authentication	0.9998 0.0002	**0.9998** **0.0002**
Blood Transfusion Service Center	0.7593 **0.3043**	**0.7598** 0.3054
Caeserian Section	**0.7583** 0.1129	0.7570 **0.1121**
Cryotherapy	0.9639 0.0165	**0.9648** **0.0149**
FourClass	0.8334 **0.1969**	**0.8335** 0.1972
Liver Disorders	0.7506 **0.3151**	**0.7512** 0.3155
Vertebral Column	0.9414 **0.0565**	**0.9429** 0.0576

We utilize w-MBD method to see how the selection of specific instances affects the performance of the algorithm. In Table 7, it is clear to see that w-MBD (12-12) dominates MBD (12-12) in terms of AUC performances in 6 of 7 datasets, however it does not show the same superiority on optimality gaps. That shows w-MBD is successful in improving the LB but not UB.

CHAPTER V

CONCLUSION

In this study, we have investigated the mixed integer optimization model which directly maximizes the area under Receiver Operating Characteristic curve for binary classification problems. We have shown that the mathematical model, the original MIP model, is a special case of MAX FS problem, which is proved to be an NP-Hard problem, where it has the same structure for all constraints and infeasibility of linear system is unknown, and it cannot be solved to optimality in reasonable time.

We have introduced several solution approaches including the reformulation of original MIP model, heuristic bounding methods that utilize McCormick inequalities and Benders decomposition approach with combinatorial cuts for original MIP model. Our solution approaches do not also provide optimal solutions, however they generally provide better objective function values and smaller optimality gaps than the original MIP model.

As a conclusion, there exists several potential research directions such as generating non-linear solution approaches for the reformulated model in 3.2.2 instead of the substitution of bilinear term in (9) and (10), and improving the quality of Benders cuts and subset selection to tighten the feasible region of master problem in order to reach optimality in a more reasonable time.

Bibliography

[1] O. Chapelle, B. Scholkopf, and A. Zien, "Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]," *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 542–542, 2009.

[2] 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.

[3] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.

[4] L. Wei, Y. Yang, R. M. Nishikawa, and Y. Jiang, "A study on several machine-learning methods for classification of malignant and benign clustered microcalcifications," *IEEE transactions on medical imaging*, vol. 24, no. 3, pp. 371–380, 2005.

[5] C. Phua, V. Lee, K. Smith, and R. Gayler, "A comprehensive survey of data mining-based fraud detection research," *arXiv preprint arXiv:1009.6119*, 2010.

[6] E. Alpaydin, *Introduction to machine learning*. MIT press, 2009.

[7] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *European conference on machine learning*, pp. 137–142, Springer, 1998.

[8] T. Fawcett and F. Provost, "Adaptive fraud detection," *Data mining and knowledge discovery*, vol. 1, no. 3, pp. 291–316, 1997.

[9] F. J. Provost, T. Fawcett, R. Kohavi, *et al.*, "The case against accuracy estimation for comparing induction algorithms.," in *ICML*, vol. 98, pp. 445–453, 1998.

[10] A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

[11] S. Rosset, "Model selection via the auc," in *Proceedings of the twenty-first international conference on Machine learning*, p. 89, ACM, 2004.

[12] L. Yan, R. H. Dodier, M. Mozer, and R. H. Wolniewicz, "Optimizing classifier performance via an approximation to the wilcoxon-mann-whitney statistic," in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 2003.

[13] C. Ferri, J. Hernández-Orallo, and R. Modroiu, "An experimental comparison of performance measures for classification," *Pattern Recognition Letters*, vol. 30, no. 1, pp. 27–38, 2009.

[14] C. Cortes and M. Mohri, "Auc optimization vs. error rate minimization," in *Advances in neural information processing systems*, pp. 313–320, 2004.

[15] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve.," *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.

[16] S. B. Kotsiantis, I. D. Zaharakis, and P. E. Pintelas, "Machine learning: a review of classification and combining techniques," *Artificial Intelligence Review*, vol. 26, no. 3, pp. 159–190, 2006.

[17] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[18] M. Bramer, *Principles of data mining*, vol. 180. Springer, 2007.

[19] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[20] J. R. Quinlan, *C4. 5: programs for machine learning*. Elsevier, 2014.

[21] L. Breiman, *Classification and regression trees*. Routledge, 2017.

[22] I. Kononenko, "Estimating attributes: analysis and extensions of relief," in *European conference on machine learning*, pp. 171–182, Springer, 1994.

[23] K. P. Bennett, "Decision tree construction via linear programming," tech. rep., University of Wisconsin-Madison Department of Computer Sciences, 1992.

[24] D. E. Brown and C. L. Pittard, "Classification trees with optimal multi-variate splits," in *Proceedings of IEEE Systems Man and Cybernetics Conference-SMC*, vol. 3, pp. 475–477, IEEE, 1993.

[25] S. K. Murthy, "Automatic construction of decision trees from data: A multi-disciplinary survey," *Data mining and knowledge discovery*, vol. 2, no. 4, pp. 345–389, 1998.

[26] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.

[27] V. N. Vapnik, "An overview of statistical learning theory," *IEEE transactions on neural networks*, vol. 10, no. 5, pp. 988–999, 1999.

[28] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[29] S. Abe, *Support vector machines for pattern classification*, vol. 2. Springer, 2005.

[30] J. Mourao-Miranda, A. Reinders, V. Rocha-Rego, J. Lappin, J. Rondina, C. Morgan, K. D. Morgan, P. Fearon, P. B. Jones, G. A. Doody, *et al.*, "Individualized prediction of illness course at the first psychotic episode: a support vector machine mri study," *Psychological medicine*, vol. 42, no. 5, pp. 1037–1047, 2012.

[31] J. Ramírez, J. Górriz, D. Salas-Gonzalez, A. Romero, M. López, I. Álvarez, and M. Gómez-Río, "Computer-aided diagnosis of alzheimers type dementia combining support vector machines and discriminant set of features," *Information Sciences*, vol. 237, pp. 59–72, 2013.

[32] V. Blanz, B. Schölkopf, H. Bülthoff, C. Burges, V. Vapnik, and T. Vetter, "Comparison of view-based object recognition algorithms using realistic 3d models," in *International Conference on Artificial Neural Networks*, pp. 251–256, Springer, 1996.

[33] X. Wang and P. M. Pardalos, "A survey of support vector machines with uncertainties," *Annals of Data Science*, vol. 1, no. 3-4, pp. 293–309, 2014.

[34] J. Wei, Z. Jian-Qi, and Z. Xiang, "Face recognition method based on support vector machine and particle swarm optimization," *Expert Systems with Applications*, vol. 38, no. 4, pp. 4390–4393, 2011.

[35] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, 2018.

[36] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," 2011.

[37] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge & Data Engineering*, no. 9, pp. 1263–1284, 2008.

[38] T. Fawcett, "An introduction to roc analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[39] U. Brefeld and T. Scheffer, "Auc maximizing support vector learning," in *Proceedings of the ICML 2005 workshop on ROC Analysis in Machine Learning*, 2005.

[40] H. Verrelst, Y. Moreau, J. Vandewalle, and D. Timmerman, "Use of a multi-layer perceptron to predict malignancy in ovarian tumors," in *Advances in Neural Information Processing Systems*, pp. 978–984, 1998.

[41] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The annals of mathematical statistics*, pp. 50–60, 1947.

[42] D. Bertsimas, A. Chang, and C. Rudin, "A discrete optimization approach to supervised ranking," in *Proceedings of the 5th INFORMS Workshop on Data Mining and Health Informatics (DM-HI 2010)*, 2010.

[43] E. Amaldi, M. E. Pfetsch, and L. E. Trotter Jr, "On the maximum feasible subsystem problem, iiss and iis-hypergraphs," *Mathematical Programming*, vol. 95, no. 3, pp. 533–554, 2003.

[44] D. M. Green and J. A. Swets, *Signal Detection Theory and Psychophysics*. New York: Wiley, 1966.

[45] E. Amaldi, M. E. Pfetsch, and L. E. Trotter, "Some structural and algorithmic properties of the maximum feasible subsystem problem," in *International Conference on Integer Programming and Combinatorial Optimization*, pp. 45–59, Springer, 1999.

[46] M. Parker and J. Ryan, "Finding the minimum weight iis cover of an infeasible system of linear inequalities," *Annals of Mathematics and Artificial Intelligence*, vol. 17, no. 1, pp. 107–126, 1996.

[47] H. J. Greenberg and F. H. Murphy, "Approaches to diagnosing infeasible linear programs," *ORSA Journal on Computing*, vol. 3, no. 3, pp. 253–261, 1991.

[48] E. Amaldi, *From finding maximum feasible subsystems of linear systems to feed-forward neural network design*. PhD thesis, Citeseer, 1994.

[49] J. W. Chinneck, *Feasibility and Infeasibility in Optimization: Algorithms and Computational Methods*, vol. 118. Springer Science & Business Media, 2007.

[50] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part iconvex underestimating problems," *Mathematical programming*, vol. 10, no. 1, pp. 147–175, 1976.

[51] F. A. Al-Khayyal and J. E. Falk, "Jointly constrained biconvex programming," *Mathematics of Operations Research*, vol. 8, no. 2, pp. 273–286, 1983.

[52] Z. C. Taşkin, "Benders decomposition," *Wiley Encyclopedia of Operations Research and Management Science*, 2010.

[53] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[54] D. Dua and C. Graff, "UCI machine learning repository," 2017.

[55] I.-C. Yeh, K.-J. Yang, and T.-M. Ting, "Knowledge discovery on rfm model using bernoulli sequence," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5866–5871, 2009.

[56] F. Khozeimeh, R. Alizadehsani, M. Roshanzamir, A. Khosravi, P. Layegh, and S. Nahavandi, “An expert system for selecting wart treatment method,” *Computers in biology and medicine*, vol. 81, pp. 167–175, 2017.

[57] F. Khozeimeh, F. Jabbari Azad, Y. Mahboubi Oskouei, M. Jafari, S. Tehranian, R. Alizadehsani, and P. Layegh, “Intralesional immunotherapy compared to cryotherapy in the treatment of warts,” *International journal of dermatology*, vol. 56, no. 4, pp. 474–478, 2017.

[58] R. O. Duda, P. E. Hart, *et al.*, *Pattern classification and scene analysis*, vol. 3. Wiley New York, 1973.

[59] J. Davis and M. Goadrich, “The relationship between precision-recall and roc curves,” in *Proceedings of the 23rd international conference on Machine learning*, pp. 233–240, ACM, 2006.

VITA

İbrahim Edhem SAKARYA graduated from Çağrıbey Anatolian High School in 2012. He received his B.S. degree in Industrial Engineering from Özyeğin University in June 2017. After graduation, he joined Master of Science program in Industrial Engineering at Özyeğin University, and has been working under supervision of Assoc. Prof. Erhun Kundakcıoğlu. His research mainly focuses on combinatorial optimization and optimization in data mining.