FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

# A Comparison of Machine Learning Techniques on the Medical Data Sets

DIPLOMARBEIT

zur Erlangung des akademischen Grades

**Diplom-Ingenieur**

im Rahmen des Studiums

**Computational Intelligence**

eingereicht von

**Omid Karami**
Matrikelnummer 1129944

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuer: **Priv.-Doz. Dr. Nysret Musliu**

Wien, 27.04.2017 ____________________ ____________________

(Unterschrift Verfasser) (Unterschrift Betreuer)

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

# A Comparison of Machine Learning Techniques on the Medical Data Sets

MASTER'S THESIS

submitted in partial fulfillment of the requirements for the degree of

**Diplom-Ingenieur**

in

**Computational Intelligence**

by

**Omid Karami**

Registration Number 1129944

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: **Priv.-Doz. Dr. Nysret Musliu**

Vienna, 27.04.2017 ________________________ ________________________

(Signature of Author) (Signature of Advisor)

# Erklärung zur Verfassung der Arbeit

Omid Karami

Wopenkastarsse 3 Stiege 4 Tür 18, 1110 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, und dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe. Weiters habe ich alle Stellen der Arbeit, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, unter Angabe der Quelle als Entlehnung kenntlich gemacht.

| | |
|---|---|
| (Ort, Datum) | Omid Karami |

# Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisor **Priv.-Doz. Dr. Nysret Musliu** for the continuous support of my master thesis, for his patience, motivation, enthusiasm, and immense knowledge. His guidance and constructive criticism helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my master thesis.

Besides my advisor, I would also like to thank my family, especially to my father and my mother, for supporting me throughout my life.

# Kurzfassung

„Maschinelles Lernen“ ist eine der interessantesten Neuentwicklungen im Bereich der Wissenschaft der Datentechnik und findet seinen Anwendungsbereich als Beispiel in der Medizin. Hier wird diese Technik insbesondere angewendet um wichtige medizinische Entscheidungen zutreffen. Mit Hinblick auf die sehr große Anzahl der gesammelten Patienten-Daten und deren sehr rasant wachsenden Mengen, ist die korrekte Ausarbeitung bzw. Verständnis dieser Unmengen an Daten durch einen Menschen fast nicht mehr möglich und kann zu fatalen gesundheitlichen Entscheidungen führen.

Deshalb beschäftigt sich diese Diplomarbeit mit dem Thema Maschinelles Lernen und seinen Einsatz im Bereich der Medizin. Hier existiert aber eine Vielzahl an Techniken für die Bearbeitung der Daten und die Herausforderung wird sich hier bei der Auswahl der richtigen Technik zur Bearbeitung der spezifischen Daten stellen. Obwohl für jeden Teilbereich der Medizin mehrere Auswertung-Techniken bereits existieren, ist es aber noch nicht klar, ob man bei der Auswahl anderer Maschinelles- Lernen-Methoden eine Verbesserung der Ergebnisse erreichen kann.

Die Quelle der benutzten Daten in dieser Diplomarbeit ist das UCI-Repository in Hinblick auf schwer bzw. nicht genau klassifizierbare Datensätze. Als erstes wurden die existierenden Daten-Analyse und deren Ergebnisse durch die Anwendung verschiedene Maschinelles-Lernen-Methoden durchleuchtet. Basierend auf den Ergebnissen dieser Voruntersuchung wurden neuere/erweiterte bzw. andere Klassifikatoren für die Daten angewendet. Die dazugehörigen Parameter der Klassifikatoren wurden aus verschiedenen Konfigurationen experimentell herausberechnet und auf die ursprünglichen Daten angewendet. Auch Auswirkungen der Vorab-Bearbeitung und Vorab-Analyse der Daten auf die Endergebnisse wurden untersucht.

Die Ergebnisse zeigen, dass die Benutzung der richtigen Daten-Vorab-Analyse und Einstellung der Parameter für die Maschinelles Lernen Algorithmus sich wesentlich auf ein gutes und korrektes Ergebnis auswirken. Obwohl die besten und genaueren Ergebnisse durch die Anwendung verschiedene Maschinelles-Lernen-Algorithmen erzielt worden sind, haben unsere Untersuchungen aber gezeigt, dass „AdaBoost“ und „random forest“ gute Resultate liefern können.

# Abstract

Machine learning is one of the most interesting topics of research that is applied in many domains such as for example medicine. It is going to play an important role for decision support in this area. The amount and complexity of recorded data in this area increases constantly, which makes it harder for humans to make right decisions that are important for human lives.

The focus of this thesis is the application of the machine learning techniques in medical data. An important question which arises, when applying machine learning techniques, is the selection of the most suitable techniques for a specific application. Although many researchers compared different techniques for specific medical domains, often it is not clear if the results for these domains can be still improved by applying other machine learning techniques.

In this thesis, several medical data sets were selected from UCI repository. The focus was particularly on data sets for which is not easy to achieve high classification accuracy. In this thesis we first reviewed the machine learning techniques which have been used for the selected data sets and analyzed the existing results. We then experimentally evaluated various new classifiers on these data sets. The parameters in each classifier were investigated and experiments with various configurations were performed. Furthermore, we evaluated the impact of the preprocessing techniques on selected datasets.

The experiments showed that the use of preprocessing techniques and parameter tuning is very important to achieve good performance for the most machine learning algorithms. Although the best results were obtained by various machine learning algorithms, our experiments showed that ensemble learning algorithms such as AdaBoost and random forest gave usually good results.

# Contents

# Table of Figures

# Chapter 1

# Introduction

In these days, the amount and complexity of recorded data is incredibly increased in many domains such as for example in medicine. Most of the devices, which are used in the medical area have not only the ability to record the patient status, but also to provide digital output in the local or central storage management system including images, test results, patient records, history of diagnoses, etc [Rensimer et al.2000]. This information could be valuable for various purposes like diagnosis, reducing therapy cost, discovering new medical hypotheses, provide evidence of proposed hypotheses, helping healthcare organizations to improve deficiencies, and so forth ([Cios et al.2002] & [Sim et al.2001]).

Machine learning (ML) techniques are used in different areas to analyze the data and find hidden patterns in the data. One of those areas is medicine, where machine learning techniques have been used very successfully for Clinical Decision Support Systems [Berner 2007] like health information system or new health-care system ([Yoo et al. 2012] & [Parvez et al. 2015]).

This thesis will focus on a variety of classification problems in medicine such as the classification of a liver patient from a non-liver patient. Researchers have used previously different techniques for such tasks including classifiers like k-nearest neighbor [Aha et al. 1991], support vector machine [Vapnik 1995], Decision trees [Quinlan 2014], random forest [Breiman 2001], etc.

An important question when applying machine learning techniques is the selection of the most suitable techniques for a particular application. Although many researchers compared different techniques for particular medical domains, often it is not clear if the results for these domains can be still improved by applying other machine learning techniques.

## 1.1 Aim of the Master’s Thesis

The aim of this thesis is to extensively compare different machine learning techniques on well-known medical data sets. In this thesis we will deal with these questions:

- Which techniques give best results for the selected medical datasets?
- Which is the impact of different preprocessing techniques on the prediction accuracy?
- Which is the impact of parameter configuration on the performance of algorithms for these datasets?
- Which lessons can we draw from experimental results and studied works?

## 1.2 Results of the Master’s Thesis

The main results obtained by this master’s thesis are:

- We selected 8 suitable medical data sets and analyzed related literature and results for these selected data sets.

- We tested and evaluated various machine learning techniques on selected data sets. In the experiment various configurations of classifiers and different preprocessing techniques were used and evaluated.

- We compared the results to the existing results in the literature on each data sets. In some cases better result could be obtained.

## 1.3 Structure of the Master’s Thesis

The rest of this thesis is organized as following. **Chapter 2** gives a background and an overview of machine learning techniques that were used in the medical domain. In **Chapter 3** an overview of medical data sets that were used in this thesis is given. **Chapter 4** provides a literature overview of selected data sets and present detailed experimental results for different classifiers. In **Chapter 5** a summary of results obtained in chapter 4 is given and the applied techniques are compared. The conclusions are given in **Chapter 6**.

# Chapter 2

# Machine learning techniques

This chapter briefly explains different machine learning techniques. The term machine learning was defined by Samuel in 1959 as “a field of study that gives computers the ability to learn without being explicitly programmed.” [Samuel 1959]. Later on, the more precise definition was presented by Mitchell in 1997 as “a computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E” [Mitchell 1997]. Machine learning has become one of the most popular technologies over the past two decades. According to mentioned researchers, there are different learning strategies. One of these includes learning from a training data set. Based on the strategy the different machine learning techniques are classified in rule induction, neural networks, case-based reasoning, genetic algorithms, inductive logic programming [Bose et al. 2001] etc.

Many different learning methods and techniques were proposed by researchers in the machine learning area in the past two decades. Such as Induction of rules [Clark et al.1989], k-nearest neighbour [Aha et al. 1991], Support Vector Machine [Vapnik 1995], decision trees [Quinlan 2014], Naive Bayes and Bayesian networks [John et al. 1995], meta-methods like bagging [Breiman 1996] and AdaBoost [Freund et al.1997], multilayer perceptron [Gardner et al.1998], random forest [Breiman 2001], etc.

An overview of the characterization of machine learning techniques and their application on various data sets with different operational characterizations was elaborated in [Bose et al. 2001].

In this thesis we applied: preprocessing (normalization; standardization; discretization and dealing with missing values), classification (k-NN; decision trees; naive Bayes, random forest; bagging; AdaBoost; SVM) and different performance measurements. Figure 2.1 shows the process of using machine learning in medical applications.

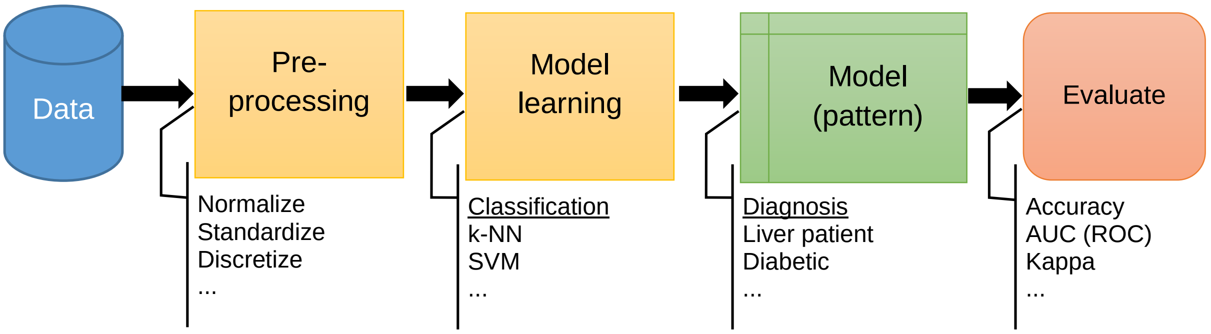


*Figure 2.1- The process of machine learning in a medical applications*

## 2.1 Classifiers

One of the fundamental tasks in data mining is classification. Classification is vastly used in human assistant intelligent applications, since its basic functionality is to assign a class label for the given instances [Pawel 2015]. In this thesis, k-NN, decision tree, naive Bayes, random forest, bagging, AdaBoost and SVM are chosen for investigation purpose. The different parameters of selected classifiers were evaluated and compared. The result shows the most impactful parameters of these classifiers. During the evaluation the input data were not preprocessed to focus solely on the classifiers behavior. However the preprocessing could be done in order to improve the end result.

### 2.1.1 *K-nearest neighbors algorithm*

It is one of the simplest machine learning algorithms that can be used for classification and regression. This algorithm belongs to the class of lazy learners [Han et al.2006]. The main concept of k-NN is to give an output for the new examples based on k nearest neighbors. The algorithm compares the given example with instances in the training dataset and similar ones which are found. In this algorithm, the training dataset will be mapped to a n-dimensional space (n describes the number of features). The algorithm searches the k closest neighbor samples in the mapped space. The parameter k determines the number of neighbors that should be concerned. The closest neighbor is defined via the distance function. Finally, the unknown class value is predicted according to its priority to the most common class value in the k nearest neighbors.

Distance function measures the distance between two instances using a defined metric [Linoff et al.2011]. Different distance metrics are already used for the measurement, which appeared in Equation 2.1 to Equation 2.3, euclidean, manhattan and minkowski respectively.

$$d(x_1, x_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2} \qquad \text{Equation 2.1}$$

$$d(x_1, x_2) = \sum_{i=1}^{n} |x_{1i} - x_{2i}| \qquad \text{Equation 2.2}$$

$$d(x_1, x_2) = \left( \sum_{i=1}^{n} (|x_{1i} - x_{2i}|)^q \right)^{1/q} \qquad \text{Equation 2.3}$$

Weighting function (distance weighted in k-NN [Dudani 1976]) allows the classifier to assign a higher importance to closer neighbors in order to predict the class label. It is predicted by the weighting function (w) in Equation 2.4 and Equation 2.5 which adds weighted distance i to neighbor class i.

$$w = \frac{1}{d(x_1, x_2)}, \qquad w = 1 - d(x_1, x_2) \qquad \text{Equation 2.4}$$

$$perdict\ class = \sum_{i=1}^{k} w_i C_i \qquad \text{Equation 2.5}$$

An example of k-NN is given in Figure 2.2 to predict the class value of “star”. The result is labeled as “Triangle” for k=3 and “Circle” for k=5.

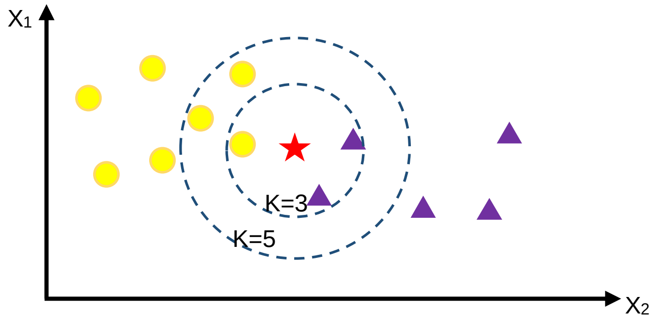


*Figure 2.2- An example of k-NN algorithm to predict a new sample with two attributes [Mitchell 1997]*

The parameter k, distance, and the weighting function (which are based on distance) were selected as the most important factors in the k-NN classifier. In this thesis the euclidean distance (Equation 2.1) was applied as the distance function. The k parameter was varied from 1 to the numbers of the samples.

## 2.1.2 *Decision tree C4.5 (j48)*

The algorithm constructs a decision tree which is developed by Quinlan in 1993 based on its ID3 algorithm [Quinlan 2014]. It deals with continuous and discrete attributes. The algorithm creates a decision tree based on a training dataset. In each level the attribute is selected based on maximal information gain. Figure 2.3 presents a simple example of this algorithm. The algorithm used the X axis for the first decision and the Y axis for the last decision making.

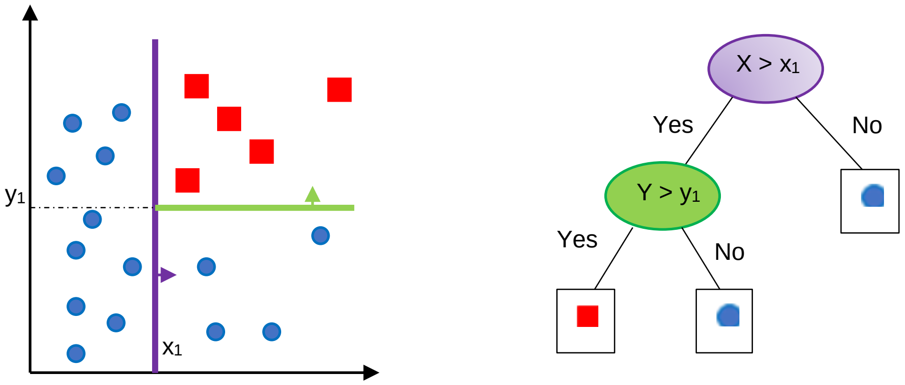


*Figure 2.3 - An example of a decision tree using the C4.5 algorithm [Witten et al.2005]*

In this thesis J48 was used, which is an equivalent classifier for the C4.5 algorithm in Weka.

## 2.1.3 *Naive Bayes*

This algorithm is the simplest Bayesian classifier based on Bayes theorem [Russell et al.2009]. It's also called simple Bayesian or independence Bayes [Han et al.2006], and it was introduced in the early 1960s. The classifier assumes the values of attributes are independent of each other (conditional independence) given the data. This feature makes the classifier simpler than other Bayes classifiers [Pawel 2015]. The main idea of the algorithm is to assign a label to the new examples based on a higher probability class. The algorithm compares the probability of classes for new examples under the given training data set. A vector of X = $(x_1, x_2, \dots, x_n)$ defined based on the n attributes (independent variables). $P(C_k|X)$ shows the probability of k possible class label for given vector. The algorithm compares probabilities of class labels based on the given example:

$$P(C_i|X) > P\left(C_j|X\right) \;\; for \;\; 1 \leq j \leq m \;\; (class\; lables), j \neq i \qquad \text{Equation 2.6}$$

The probability is computed using Bayes' theorem

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \qquad \text{Equation 2.7}$$

As we are dealing with independent attributes, the computation of $P(X|C_i)$ can be done as following

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) = P(x_1|C_i) \times P(x_2|C_i) \times \dots \times P(x_n|C_i) \qquad \text{Equation 2.8}$$

In Figure 2.4, the schema of naive Bayes is shown in the left side and an example model of playing tennis in the right side.

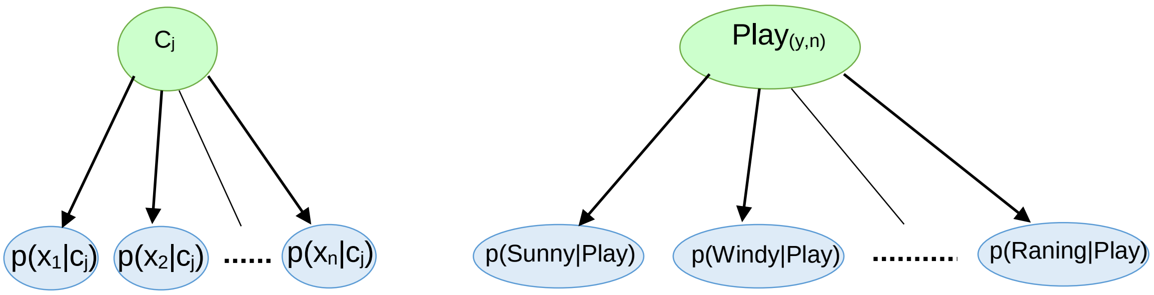


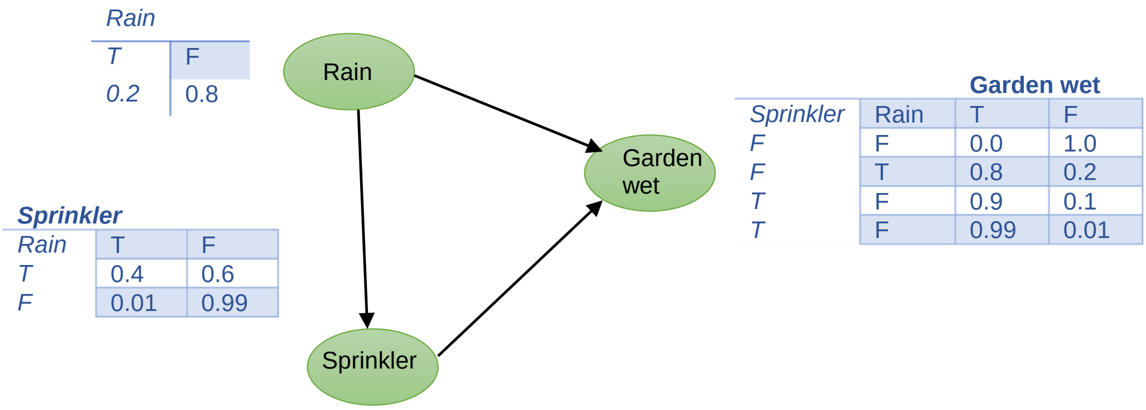
*Figure 2.4 – The schema of naive Bayesian classifier [Witten et al.2005]*

### 2.1.4 *Bayesian networks*

This algorithm is based on Baye's theorem [Han et al.2006]. In comparison to the naive Bayes, there are some dependencies between attributes. The algorithm provided a direct acyclic graph model of relationships between attributes which affected the learning probabilities. Each attribute has a conditional probability table which specifies the probability of conditional distribution on its parent. The main idea of the algorithm is to show dependency using edges in the graph. The meaning of missing an edge is conditional independence in the graph. After the acyclic graph is constructed, the conditional probability tables are specified from the training dataset. The joint probability density function of the Bayesian network is defined as follow:

$$P(X) = \prod_{i \in V} P\left(x_i | x_{parents(x_i)}\right) \qquad \text{Equation 2.9}$$

Figure 2.5 provides a schema of an example of Bayesian network with its conditional probability tables.



Rain

| T | F |
|---|---|
| 0.2 | 0.8 |

Sprinkler

| Rain | T | F |
|---|---|---|
| T | 0.4 | 0.6 |
| F | 0.01 | 0.99 |

Garden wet

| Sprinkler | Rain | T | F |
|---|---|---|---|
| F | F | 0.0 | 1.0 |
| F | T | 0.8 | 0.2 |
| T | F | 0.9 | 0.1 |
| T | F | 0.99 | 0.01 |

*Figure 2.5 – A simple example of Bayesian network [Muhammad et al.2015]*

### 2.1.5 *Random forest*

The algorithm was proposed first by [Breiman 2001]. The main idea of this algorithm is to use a collection of an arbitrary number of simple trees, each can predict an independently class value. The final prediction is taken from the prediction average for the given data. The randomization is used for choosing features in the tree creation process. Each tree has a bootstrap sample from an original dataset with replacement. This algorithm is used for both classification and regression problems [Pawel 2015].

An example of random forest is given below to predict the class value of c in Figure 2.6. The result is predicted by an average of all generated trees using Equation 2.10.

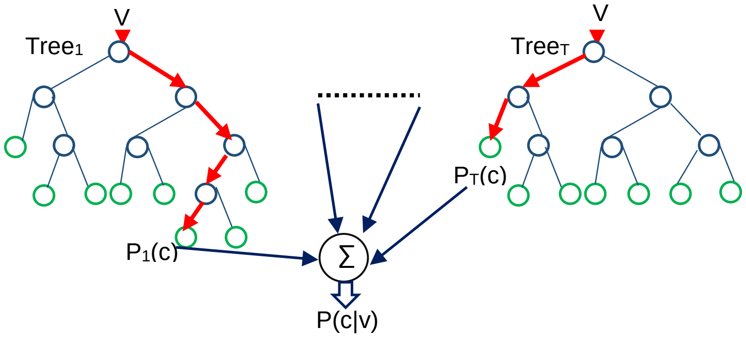


*Figure 2.6- An example of random forest algorithm*

$$P(c|v) = \frac{1}{T}\sum_{i=1}^{T} P_i(c|v) \qquad \text{Equation 2.10}$$

There are some parameters which usually impact on the performance of the algorithm. The number of trees in the forest is one of these parameters. A better result is obtained usually by using a larger number of trees in the forest. There is also the possibility of over-fitting here. The default value of 100 for number of trees has been chosen by Weka. The maximum number of features per split is also an important parameter during the tree creation. The default value has been chosen as square root of the feature numbers (by Weka). This parameter has a large impact on the behavior of the algorithm. The height (or maximum depth) in each tree is another parameter in the tree creation process. The full size is used as the default value in Weka.

The number of features, number of trees and maximum tree depth size have been selected as the most significant parameters in the random forest classifier. The number of features depends on the data set, in our experiment we have used 10% to 100% of features. For the number of trees we experimented 5 to 500 (10, 20, 50, 100, 200). For maximum tree depth size we also experimented 2 to 450. In each test we used two configurations for comparing, whereby one of them is always fixed as the default value.

### 2.1.6 Bagging

The name is driven from bootstrap aggregation [Breiman 1996]. The main idea of this algorithm is to subtract training dataset to n samples (with replacement), then the average of prediction is used as the result of the classifier. This algorithm is the simplest ensemble learning algorithm. It uses a simple bootstrapping technique which generates a new dataset from the original training dataset for the use in each classifier. The final result is given from the most often selected class label. This method can stabilize unstable algorithms such as decision trees [Pawel 2015].

Figure 2.7 uses a data set shape which shows different classifier results with gray color and the average result of the classifier after ten interations on the red color.

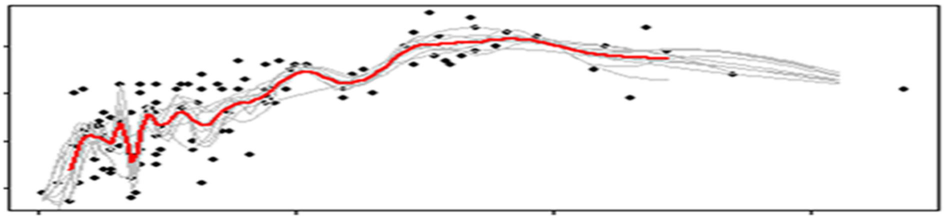


*Figure 2.7- A simple example of bagging steps process*[1]

To show the impact of the parameters in bagging, we have experimented with different base classifier (naive Bayes, k-NN, SMO, decision table and decision stump), the number of iterations of 2 to 50 (5, 10, 25, 50), and batch size of 5 to 500 (10, 20, 50, 75, 100, 200, 500) as most important parameters in the bagging classifier. In each test we used two configurations for comparing and the remaining one as default value.

[1] Picture reference: http://dmml.nu/big-data-mining-machine-learning-cloud

### 2.1.7 *AdaBoost*

The algorithm name is taken from Adaptive boosting which is introduced in 1995 by Freund and Schapire [Freund et al.1997]. The main idea of this algorithm is to build a stronger classifier by using multiple weaker classifiers which are constructed in "n" iterations. The linear combination is applied for combining multiple classifiers. In each iteration a classifier is applied and the weight of the sample in the dataset increases if it is classified incorrectly and vice versa [Pawel 2015]. Figure 2.8 illustrates a schematic example[2] of this. The top left box is an initial training dataset. The example shows four classifiers using simple decision table algorithm and in the bottom of the right box the combination of all four results is visible.

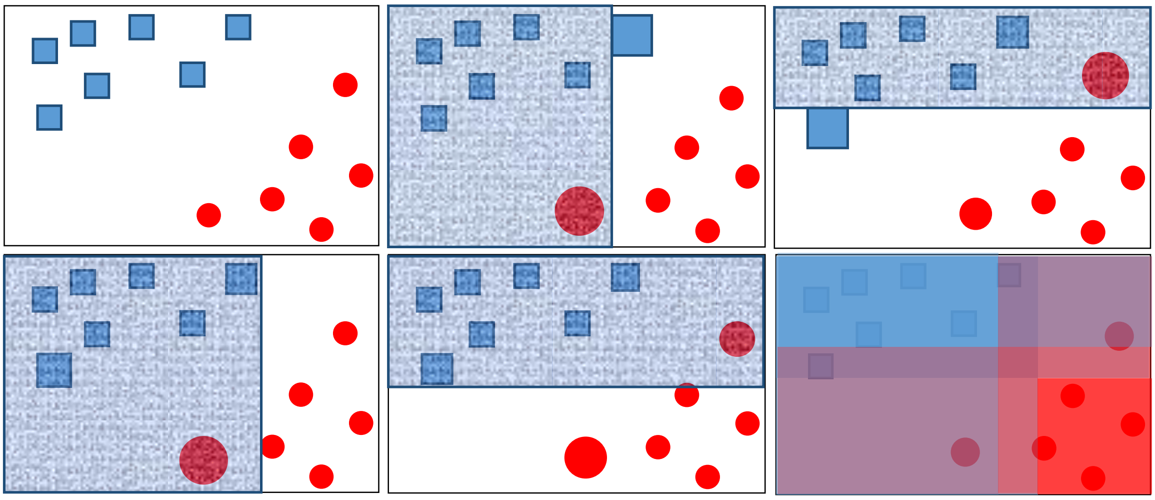


*Figure 2.8 - A schema example of AdaBoost algorithm*[2]

Regarding this algorithm, the significant parameters for evaluation are the base classifier, the weight threshold and the number of iterations.

To demonstrate the impact of parameters in the AdaBoost classifier, different base classifiers were applied (naive Bayes, k-NN, SMO, decision table and decision stump), the number of iterations was varied from 2 to 50 (5, 10, 25, 50) and the weight threshold was in range 5 to 500 (10, 20, 50, 75, 100, 200, 500). In each test two parameters were modified for the comparisons and the remaining ones were set to a fixed default value.

### 2.1.8 *SVM*

This algorithm was presented by Vladimir Vapnik in 1995 [Vapnik 1995] for solving pattern recognition problems. The algorithm maps the data in the dataset into a set of hyperplanes in higher dimensional space [Han et al.2006]. An optimal hyperplane separates existing hyperplanes in this space. The kernel functions and maximum margin separator are the most important properties of the algorithm [Han et al.2006]. Figure 2.9 illustrates schematic examples[3] of using two different groups in linear and non-linear space. On the left side there is a linear problem and on the right side, there is a non-linear problem which is transferred in higher dimension and it is converted to a linear problem.

[2] Web access: https://alliance.seas.upenn.edu/~cis520/wiki/index.php?n=lectures.boosting

[3] Web access: http://www.statsoft.com/Textbook/Support-Vector-Machines

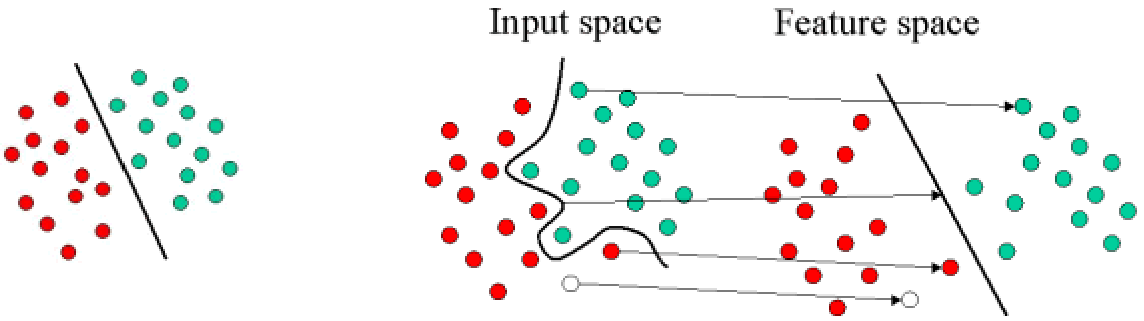


*Figure 2.9- The linear and non-linear samples in SVM[3] [Russell et al.2009]*

We used two implementation of SVM (SVM and SMO). For both of them we used different kernels. SMO is abstract name of sequential minimal optimization proposed by [Platt 1998] which is based on SVM provided by [Keerthi et al.2001].

## 2.2 Data preprocessing

In our experiment we used the following preprocessing techniques:

- **Normalization**: In this approach, the values rescale into a range between 0 and 1. This approach might be applied in the classifier where the values in the data need to be in the positive scale.

- **Standardization**: In this approach, the data values rescale to a standard normal distribution range with a mean of μ = 0 and standard deviation of σ = 1.

- **Discretization:** This technique is an important process in preprocessing phase and sometimes it is required. The general idea of this technique is to reduce the complexity or cleaning up the training dataset [Pawel 2015]. There are some classifiers which can not deal with continuous attributes. This technique can be used to convert continuous attributes to nominal or discretized values. The discretization process can be performed on one, some or all of the attributes in the training dataset. Figure 2.10 shows the simple discretization scenario for a continuous attribute. The humidity is discretized to three parts (H ≤ 40, 40 < H ≤ 80, H > 80) instead of continuous values.

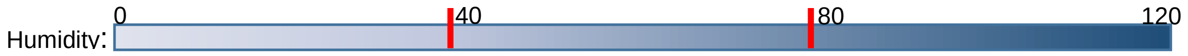
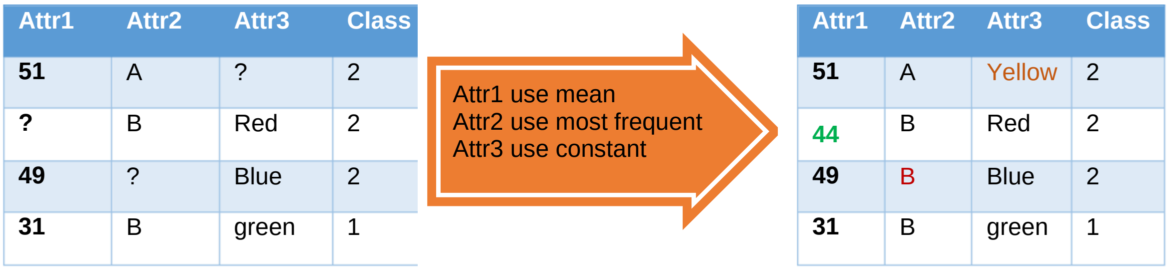


*Figure 2.10- A simple example of discretization of a continuous attribute using threshold*

- **Missing values:** There are some well-known techniques like ignoring instance, fill the missing value with a constant value, mean, average or maximum value from attribute [Han et al.2006]. Figure 2.11 shows a simple example of dealing with missing values. The attribute 1 is replaced with the mean value, attribute 2 is replaced with most frequently value and attribute 3 is replaced with constant value.



| Attr1 | Attr2 | Attr3 | Class |
|---|---|---|---|
| **51** | A | ? | 2 |
| **?** | B | Red | 2 |
| **49** | ? | Blue | 2 |
| **31** | B | green | 1 |

Attr1 use mean
Attr2 use most frequent
Attr3 use constant

| Attr1 | Attr2 | Attr3 | Class |
|---|---|---|---|
| **51** | A | Yellow | 2 |
| **44** | B | Red | 2 |
| **49** | B | Blue | 2 |
| **31** | B | green | 1 |

*Figure 2.11- An example of dealing with missing values*

In [Schafer et al. 2002], a depth overview of techniques for dealing with missing value is given.

## 2.3 Performance Evaluation

We have used Weka for evaluating of classifier performance with the 10-fold cross validation [Geisser 1993]. The k-folds cross-validation has been used for testing generated model by classifiers. Accuracy, precision, recall, confusion matrix and AUC (area under the curve of the ROC curve) are calculated using Equation 2.11, Equation 2.12 and Equation 2.13 where TP is a rate of true positive and FP is a rate of false positive [Fawcett 2006].

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \qquad \text{Equation 2.11}$$

$$Presicion = \frac{TP}{TP + FP} \qquad \text{Equation 2.12}$$

$$Recall = \frac{TP}{TP + FN} \qquad \text{Equation 2.13}$$

We have alternatively used kappa measurement for computing the classification consistency. Kappa value shows the reliability of a model generated by a classifier [Viera et al.2005]. The value is between -1 and 1. The equation is shown in Equation 2.14.

$$Kappa = \frac{Accuracy + \ P_c}{1 - \ P_c} \qquad \text{Equation 2.14}$$

$$P_c = \frac{(TP + FP) \times (TP + FN) + (FN + TN) \times (FP + TN)}{(TP + FN + FP + TN)^2} \qquad \text{Equation 2.15}$$

## 2.4 Weka

Waikato Environment for Knowledge Analysis (Weka)[4] is a set of machine learning algorithms and tools for researches for studies and analysis. The tools were developed at the University of Waikato in New Zealand and distributed based on GNU General Public License. An overview of the Weka has been provided in more details and background by [Hall et al. 2009].

In this thesis, Weka was chosen to evaluate the performance of the selected machine learning techniques using selected data sets.

[4] Web access: http://www.cs.waikato.ac.nz/ml/weka/

Explorer and Experimenter interface of the Weka has been used widely in this thesis. In the explorer, experiments can be performed with all machine learning techniques including data preprocessing, classification and so forth. Additionally there is the possibility to analyze the data and present the outcome of an experiment in a graphical interface with visualization plotting. In experimenter, there is possibility to define a set of evaluations at once.

## 2.5 Application of machine learning in medicine

This thesis focuses on classification tasks for different diseases. Machine learning techniques mentioned above have been used successfully to make predictions for various diseases including: Breast issue [Chalmers et al.2014], Diabetic retinopathy [YiNan et al.2016], Fertility [Wang et al.2014], Indian liver [Camilleri et al.2014], Mammography mass [Ferrari 2011], Spect heart [McSherry 2011], Thoracic surgery [McBride et al. 2014].

As mentioned in [Kononenko 2001], a machine learning technique utilized in medical diagnostic should contain following features: high performance, dealing with noisy data and missing values, transparency of diagnostic knowledge, explanation ability and reliability of diagnoses. The reliability of machine learning techniques and the improvement of the accuracy of classification problems have been addressed in several works by researchers [Kukar et al.2002] and it is still a challenging research topic. In [Kukar et al.2005], the reliability of medical diagnostics was studied by authors based on their previous work. In [Museli et al.2007] authors provided a framework to check consistency of machine learning outcomes.

Decision support system have become one of the bases in medical care system. In [Bates et al.2003], the authors provided a summary over eight years of their studies about decision support systems in medicine as “Ten commandments for effective clinical decision support”. In [Kuperman et al.2007], the challenges in decision support systems were addressed and authors proposed some recommendations in a related clinical decision support systems. In [Sittig et al.2008], authors reported top 10 challenges in clinical decision support systems for designing, developing, implementation ,presentation, evaluation and maintaining. Difficulties were identified and prioritized in order to develop a system successfully.

Reviews on the application of machine learning techniques in medicine are given by several researchers. One of the first reviews was given by [Lavrac 1998]. In [Kononenko 2001] a historical and a state-of-the-art view was provided. In [Harper 2005], a comparison review of classification and its performance was investigated. In [Bellazzi et al.2008], a comprehensive review of the state-of-the-art of data mining in medicine was provided. In [Yoo et al. 2012] and [Parvez et al. 2015], an overview of machine learning techniques as well as the pros and cons were given in medical and healthcare area.

# Chapter 3

# Medical Datasets

This chapter focuses on explaining the medical data sets and their properties. Medical data include images, patient interviews, patient data, ECG, EEG and etc. The amount of medical data is also increasing in terms of size and dimensions [Cios et al.2002]. The authors summarized the uniqueness of the medical data as follows:

- The collected data include various types and mostly do not have mathematical characterization which causes difficulties to provide them as a general model (pattern). Such data include images, interview results, physician observations and laboratory results.
- Incompleteness, missing or noisy values
- The privacy and security issues: those data are very sensitive and vital, especially for governments. There is always a limitation on providing data in public for research proposes.
- Statistical philosophy and special status of medicine

UCI repository[5] provides a collection of different medical datasets, which are good samples of data for researchers to evaluate the proposed techniques and methods.

In this thesis, the medical data sets that are suitable for application of machine learning techniques were identified. This was done by an intensive investigation of the medical data sets provided in UCI repository. Those data sets have been used by researchers in previous works to evaluate various machine learning methods.

In this thesis, the focus is particularly on data sets for which high performance accuracy is difficult to achieve. Data sets for which authors could easily reach 100% classification accuracy were skipped.

The following datasets have been selected from UCI repository: Breast Tissue, Cardiotocography, Diabetic Retinopathy, Fertility, Indian Liver, Mammography mass, Spect heart, Thoracic surgery.

## 3.1 Breast Tissue

The breast tissue database (BTD)[6] was provided by [M. & J. 2010]. The purpose of this data set is to classify and detect sample abnormalities and cancer in women. The data set contains 106 tissue samples collected from 64 patients using Electrical Impedance Spectroscopy (EIS) and was used in

[5] UC Irvine Machine Learning Repository: http://archive.ics.uci.edu/ml/ (Online 2016).
[6] https://archive.ics.uci.edu/ml/datasets/Breast+Tissue

[Jossinet 1996]. The EIS was performed in the range of 488 Hz to 1 MH and its outcome has been presented with nine attributes (features): IO, PA500, HFS, DA, Area, A-DA, MAX-IP, DR and P. Based on these attributes, the study has classified each tissue sample to one of the six classes “carcinoma”, “fibro-adenoma”, “mastopathy”, “glandular”, “connective”, and “adipose tissues". Figure 3.1 shows the information for the class.

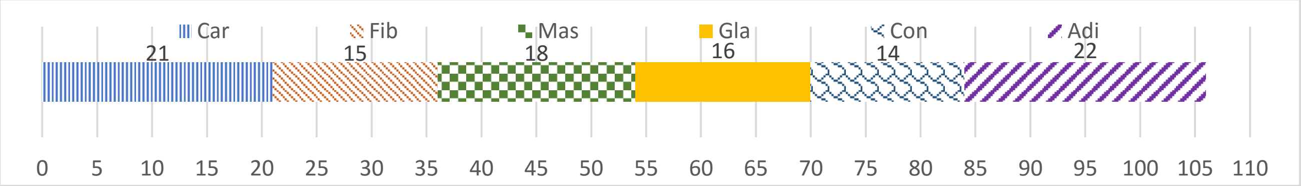


*Figure 3.1- The class labels for the breast tissue data set*

## 3.2 Cardiotocography

The Cardiotocography Data set (CTGD)[7] was provided by [Diogo et al. 2000]. The dataset contains 2126 instances which were recorded from a study at the University of Porto. It contains twenty-one attributes. Based on these attributes, each record of the dataset has been assigned to one of the ten decision classes (or second version of three-classes). Figure 3.2 shows the class labels for this dataset. It is obvious that the dataset is imbalanced in term of class label distribution.

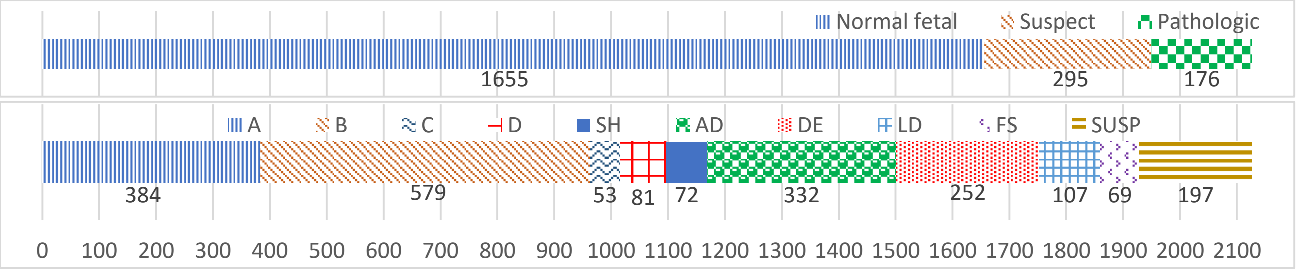


*Figure 3.2- The class labels for the CTGD*

## 3.3 Diabetic Retinopathy

The Diabetic Retinopathy Debrecen Dataset (DRDD)[8] was provided by [Antal et al. 2014] based on the Messidor images. The aim of the dataset is to predict severity of diabetic signs in Messidor images. The dataset contains 1151 instances. It contains nineteen extracted attributes. Based on these attributes, each instance of the dataset was assigned to one of the class labels: “No signs” and “signs of DR”. Figure 3.3 shows the class labels for this dataset. 540 (46.9%) of the instances are in the class No-signs and 611 (53.1%) in the class DR-signs.

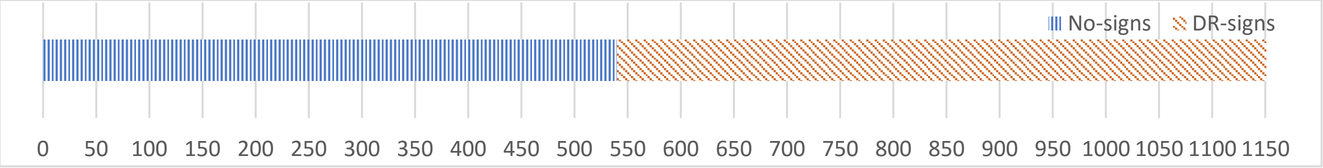


*Figure 3.3- The class labels for the Diabetic retinopathy data set*

[7] https://archive.ics.uci.edu/ml/datasets/Cardiotocography
[8] https://archive.ics.uci.edu/ml/datasets/Diabetic+Retinopathy+Debrecen+Data+Set (online on 2017)

## 3.4 Fertility

The Fertility data set (FD)[9] was provided by [Gil et al.]. The purpose of this dataset is to predict the seminal quality by using the environmental factors and lifestyle. The data set contains 100 instances of volunteers. The provided semen samples have been analyzed according to the WHO guide (World Health Organization) in the study [Gil Mendez et al.2012]. It contains nine attributes: season, age, childhood diseases, serious trauma, surgical intervention, high temperature, the frequency of alcohol consumption, smoking and the sitting hours per day. Based on these attributes, the study has classified each sample to one of the class labels: Normal and Altered. The dataset is imbalanced which is shown in Figure 3.4.

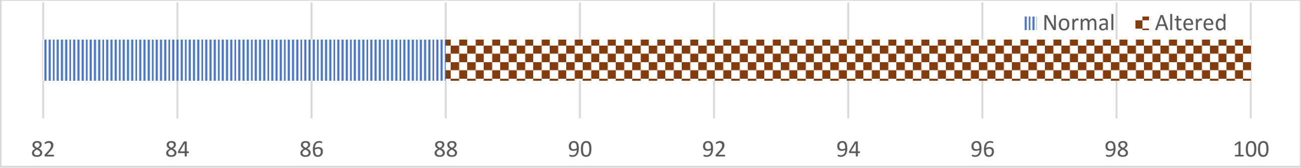


*Figure 3.4- The class labels for the Fertility data set*

## 3.5 Indian Liver

The Indian Liver Patient Dataset (ILPD)[10] was provided by [R&B&V] [Ramana et al.2012]. The dataset contains 583 instances which were recorded from the northeast of India divided by 441 male patient and 142 female patient records. It contains ten attributes: age, gender, total bilirubin, direct bilirubin, total proteins, albumin, albumin and globulin ratio, alamine aminotransferase (SGPT), aspartate aminotransferase (SGOT) and alkaline phosphatase. Based on these attributes, each record of the dataset was assigned to one of the class labels: “Liver Patient” and “Non-Liver Patient”. Figure 3.5 shows the class labels for this dataset. 72% (416) of the instances are in the class Liver Patient and 28% (167) in the class Non-Liver Patient. It is obvious that the dataset is imbalanced in term of class labels and also gender distribution.

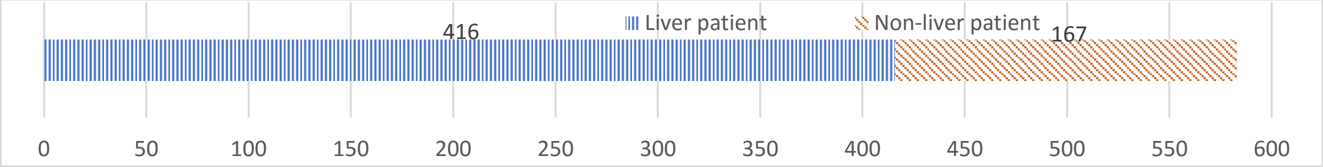


*Figure 3.5- The class labels for the Indian liver data set*

## 3.6 Mammography mass

The Mammographic Mass Dataset (MMD)[11] was provided by [E. & S., 2007]. The aim of the dataset is to predict severity of a Mammographic mass wound. The dataset contains 961 instances which were collected at the institute of Radiology of the University Erlangen Nuremberg (2003-2006). It contains five attributes: patient's age, BI-RADS assessment, shape, margin and density of BI-RADS. Based on these attributes, each instance of the dataset was assigned to one of the class labels: “Benign” and “Malignant”. Figure 3.6 shows the class labels for this dataset. 516 (54%) of the

[9] https://archive.ics.uci.edu/ml/datasets/Fertility
[10] https://archive.ics.uci.edu/ml/datasets/ILPD+%28Indian+Liver+Patient+Dataset%29
[11] https://archive.ics.uci.edu/ml/datasets/Mammographic+Mass

instances are in the class Benign and 445 (46%) are in the Malignant. There are 131 instances with missing one or two attributes.

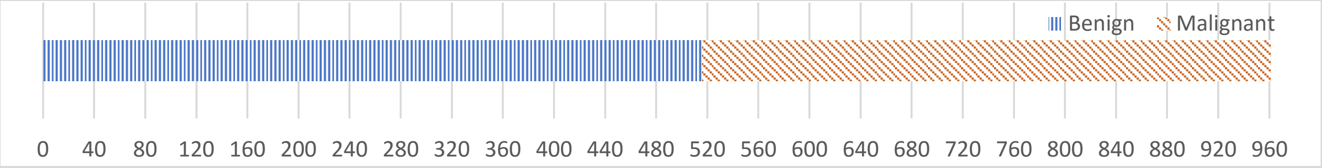


*Figure 3.6- The class labels for the Mammography mass data set*

## 3.7 Spect Heart

The SPECT[12] Dataset was provided by [Kurgan et al. 2001]. The aim of the dataset is to predict severity of Normality or abnormality of patient samples. The dataset contains 267 instances. It contains twenty-two binary attributes which were extracted from 44 continuous feature patterns from sampled images of the patients. Based on these attributes, each instance of the dataset has been assigned to one of the class labels: “Normal” and “Abnormal”. Figure 3.7 shows the class labels for this dataset. 55 (21%) of the instances are in the class Normal and 212 (79%) are in the class Abnormal. Additionally it shows the obvious imbalance of the dataset.

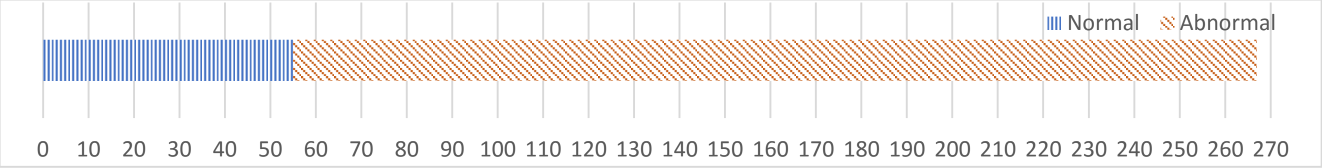


*Figure 3.7- The class labels for the SPECT heart data set*

## 3.8 Thoracic surgery

The Thoracic Surgery Dataset (TSD)[13] was collected between years 2007-2011 at Wroclaw Thoracic Surgery Center from lung cancer patients [Maciej et al. 2013]. The aim of the dataset is to predict severity of death in lung cancer patients. The dataset contains 470 instances. It contains seventeen attributes. Based on these attributes, each instance of the dataset was assigned to one of the class labels: “Death” and “Survival”. Figure 3.8 shows the class labels for this dataset. 70 (14.9%) of the instances are in the class Death and 400 (85.1%) are in the class Survival. Figure 3.8 shows the classification ratio of classes in this dataset and also that the dataset is unbalanced.

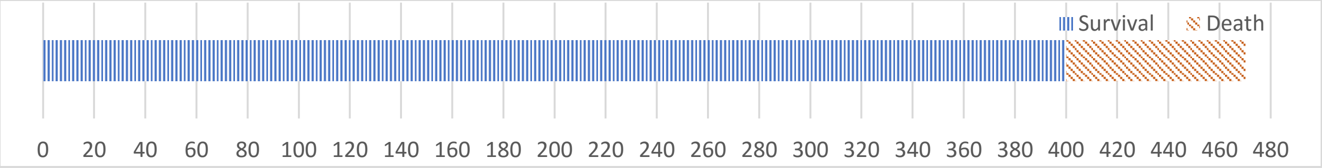


*Figure 3.8- The class labels for the Thoracic surgery data set*

[12] https://archive.ics.uci.edu/ml/datasets/SPECT+Heart
[13] https://archive.ics.uci.edu/ml/datasets/Thoracic+Surgery+Data

# Chapter 4

# Experimental Evaluation of Machine Learning Techniques

This chapter reports on experiments with selected medical data sets. To get familiar with data sets, first the related literature about data sets was investigated. Experiments were performed with various machine learning techniques. The details for each data set are given further in this chapter.

## 4.1 Breast Tissue

The BDT was used by several other researchers. The classification algorithms Cart and C4.5 were applied in [H., 2012], respectively with 70% and 67% classification accuracy. In [Y. & H., 2011] the proposed method Clustering Co-Index (an unsupervised feature selection framework has also applied using micro-averaged and macro-averaged methods) was used to classify, with 70% and 66% classification accuracy. A method using nearest-prototype style classifier was used in [Chalmers et al.2014]. This technique was optimized by a genetic algorithm in order to perform high predict positive value (PPV). The linear discriminant analysis was used in [Silva et al.2000] with 66% classification accuracy. In [D., 2015] and [Nonte 2013], authors proposed a combination of several extreme learning machine (ELM) using SVM classifier. In these studies, the scalar feature selection method was used to select the most important ones and rank the features. Various neurons in the hidden layer were applied in each ELM, and the outcomes were combined using SVM, resulting in 88% and 80% classification accuracy respectively. In [Li et al., 2012] the Semi-supervised Locality Discriminant Projection with different kernels was used for classification, which has received the help of Kernel trick to improve nonlinear classification. The method has obtained 82% classification accuracy. Based on the papers that were found for BTD , best results were obtained by [D., 2015] using ELMs and SVM classification accuracy of 88% followed by [Li et al., 2012], which used the proposed technique with an accuracy rate of 82%.

In this thesis, we additionally applied ensemble techniques and meta-classifiers like bagging, random forest, cost-sensitive learning, vote, and AdaBoost. Also the impact of parameters for chosen classifiers and different preprocessing techniques were evaluated. Figure 4.1 shows the best achieved results for used classifiers.

For each classifier the best result was taken. Different parameters were tried for tuning the classifier techniques. Regarding classification accuracy, the best accuracy of 76.5 was achieved by using AdaBoost technique, which used following configuration: SMO as base classifier with the help of polynomial kernel and a weight threshold of 100. The data was standardized in the preprocessing phase.

The second highest classification accuracy was obtained by bagging and random forest classifier. Regarding Kappa statistic and average area under the curve (ROC), all four meta-classifiers gave similar results.

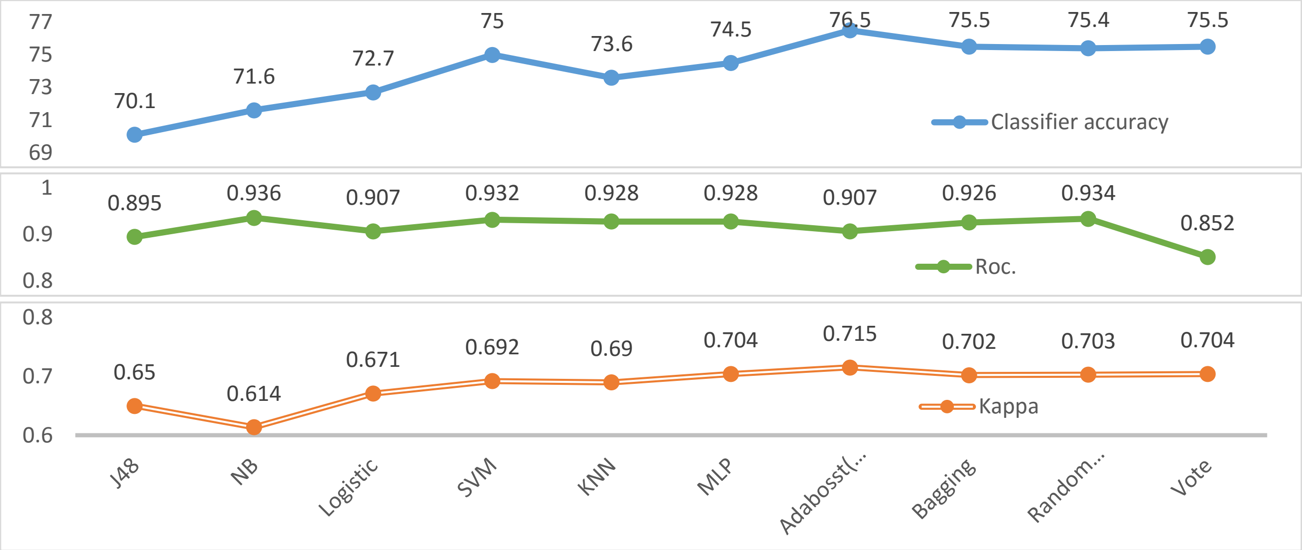


*Figure 4.1- The best achieved performance for Breast Tissue dataset using implementation of various classifiers in Weka*

We also experimented with different preprocessing techniques (discretization, feature selection, numeric to nominal, normalization, standardization) and tried to tune classifiers by using different of the configurations for selected parameters which are presented and discussed briefly in following.

Regarding the evaluation of k-NN classifier, the highest classification accuracy was obtained using:

- Weighting function (1/distance in Equation 2.4)
- k = 3

Figure 4.2 shows the evaluated performance of k-NN classifier for BT using different k value against a different weighting function. It is obvious from the results that the weighted distance using 1/distance in Equation 2.4 has fixed the wrong configuration for k.

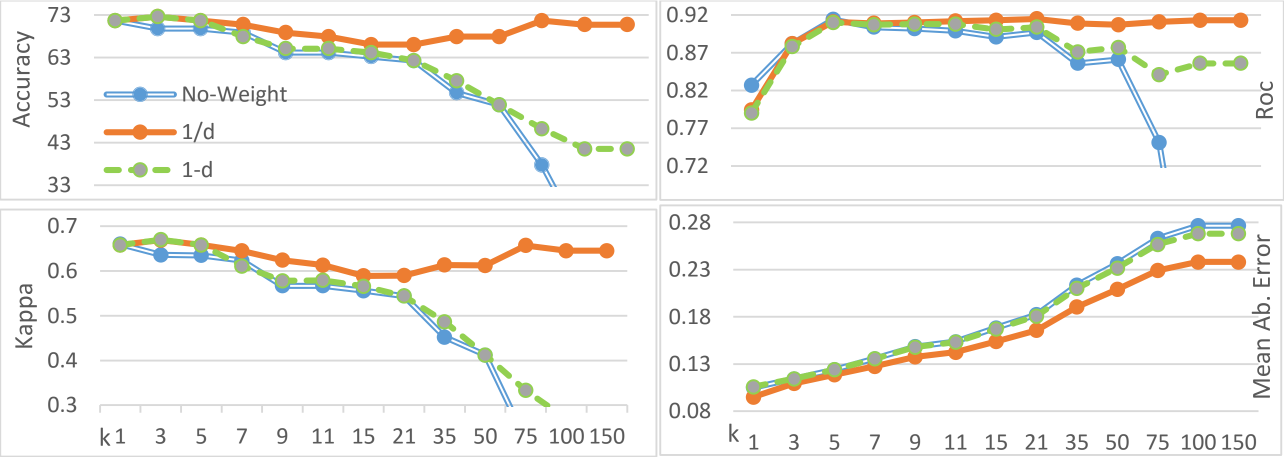


*Figure 4.2- The impact of k and weighting function in the k-NN algorithm*

Considering the results using Random Forest, the highest classification accuracy was obtained by using:

- 10% number of features
- 100 for number of trees
- Full tree depth size.

In Figure 4.3, the performance of this classifier using a different number of features against a different number of trees was shown. The results indicate that the classifier performed better with 10% of the features which means only one feature is used in every tree construction. When using more features, better result are obtained with a lower number of trees.

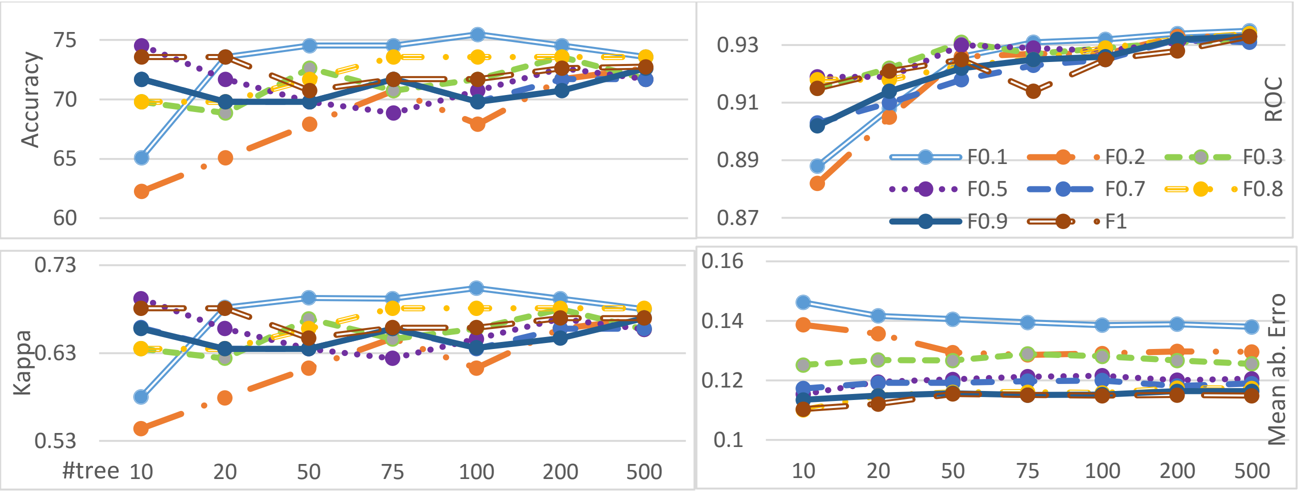


*Figure 4.3- The impact of the number of features against number of trees in the random forest*

Figure 4.4 shows the results of the algorithm with various configurations. The algorithm obtained the best results by using tree depth size = 5.

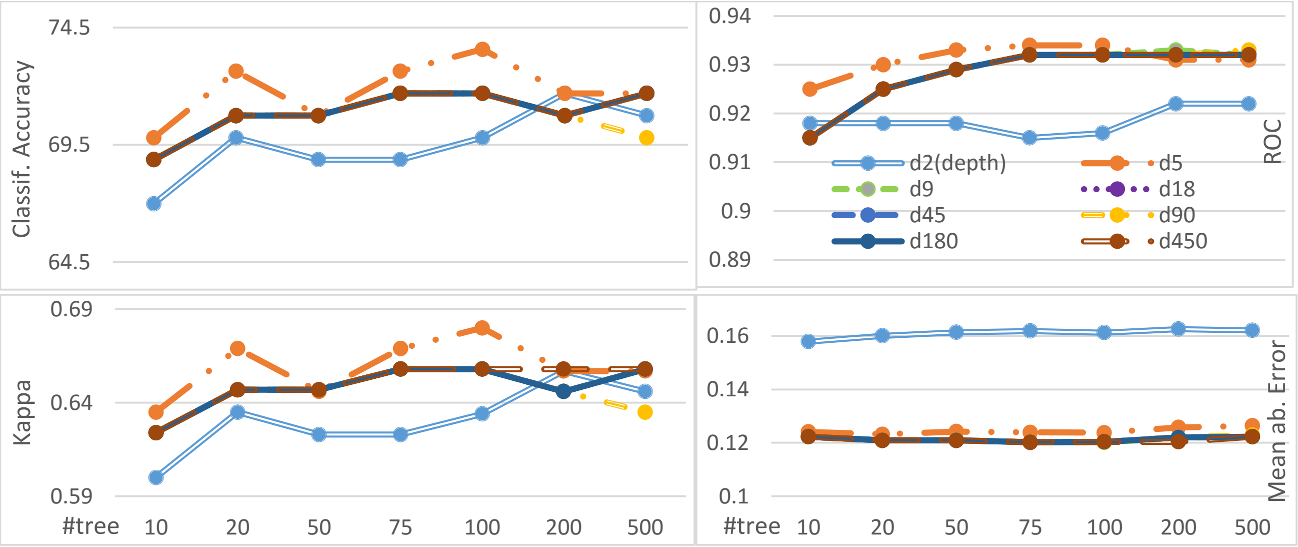


*Figure 4.4- The impact of the depth size of the tree against number of trees in the random forest*

Figure 4.5 shows the impact of the number of features and tree depth size of the results of random forest. In this figure it is obvious that a depth size higher than 18 has no effect and the best depth size is almost around 18.

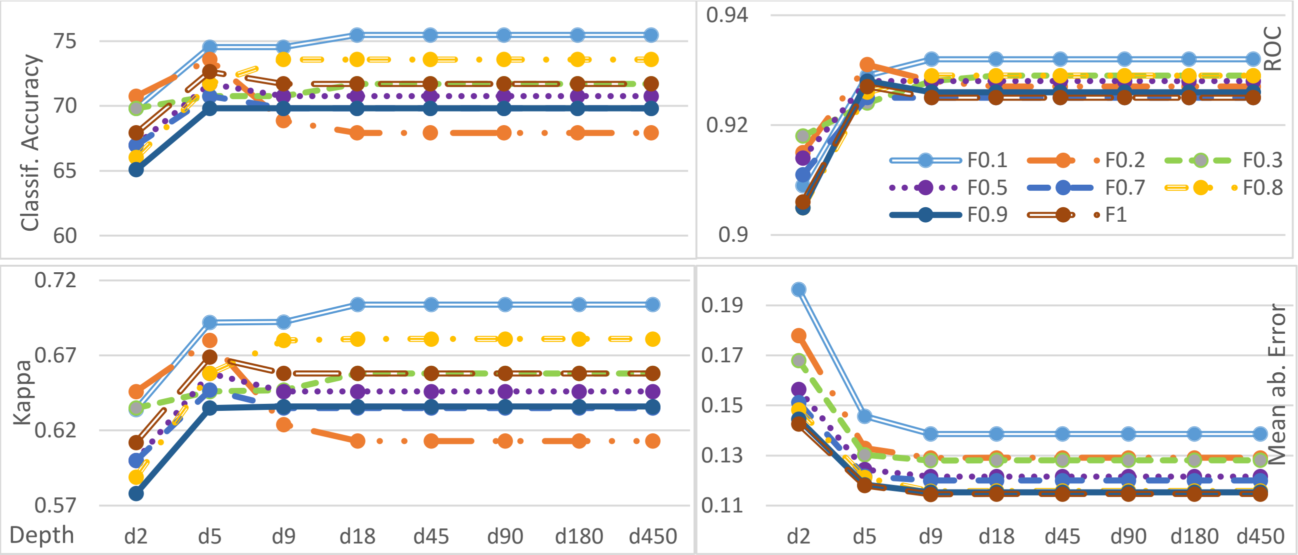


*Figure 4.5 – The impact of the number of features against the depth size of tree in random forest algorithm*

Considering the results using bagging, the highest classification accuracy was obtained by using:

- k-NN classifier
- Bag size of 70 and bag size of 500
- 50 iterations

Figure 4.6 presents the performance of the classifier. The results show that for a bag size of 70, a good performance was achieved even with a lower number of iterations. It also shows that the decision stump classifier had bad performance. In contrast the k-NN and SMO were near to each other with a higher performance outcome. k-NN was also faster than SMO.

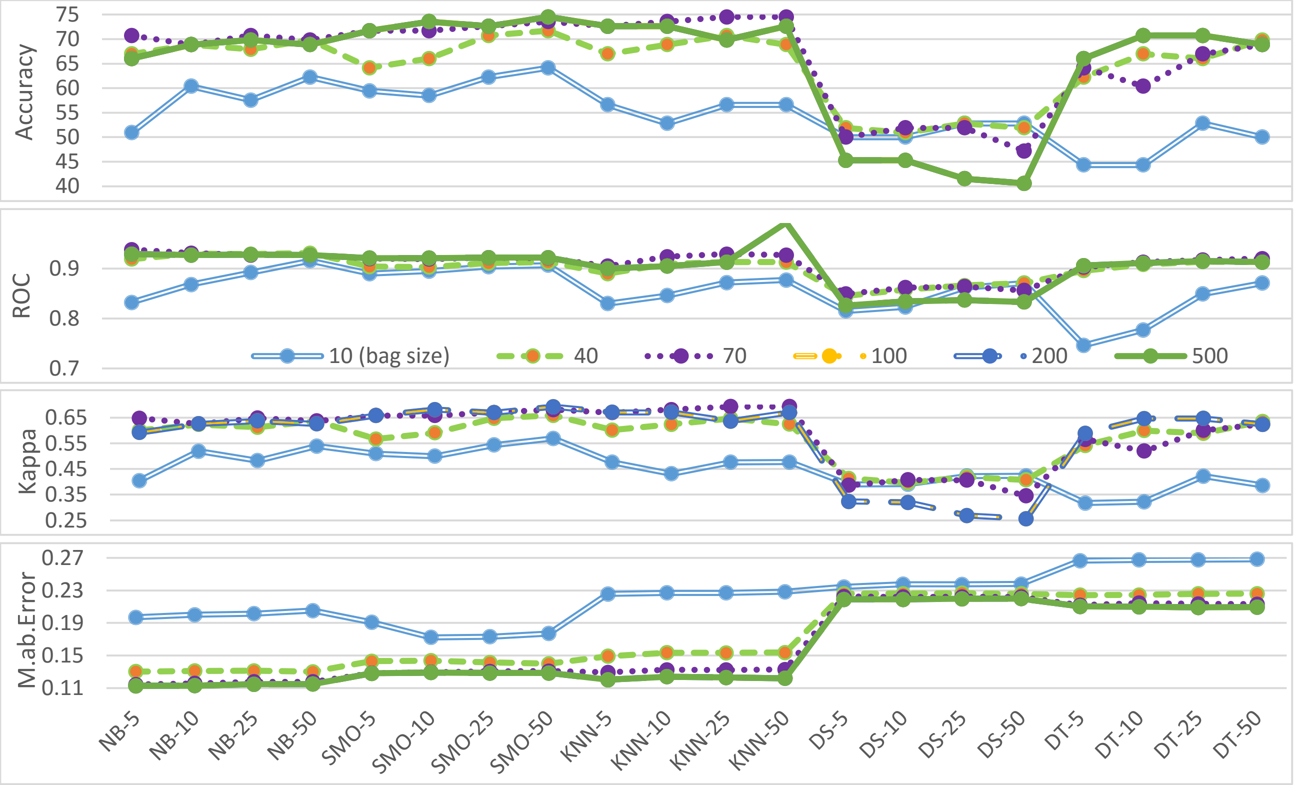


*Figure 4.6- The impact of selected parameters in the bagging algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the evaluation of AdaBoost, the highest classification accuracy was obtained by using:

- SMO and Naïve Bays classifier
- Weight threshold of 100 and more
- 5 iterations

Figure 4.7 shows the performance of the classifier with different base classifiers and iteration numbers. The results indicate that the base classifier had the highest impact in the performance. The best accuracy in this experiment was obtained by using SMO and the worst with decision stump. The number of iterations and weight threshold did not affect the performance.

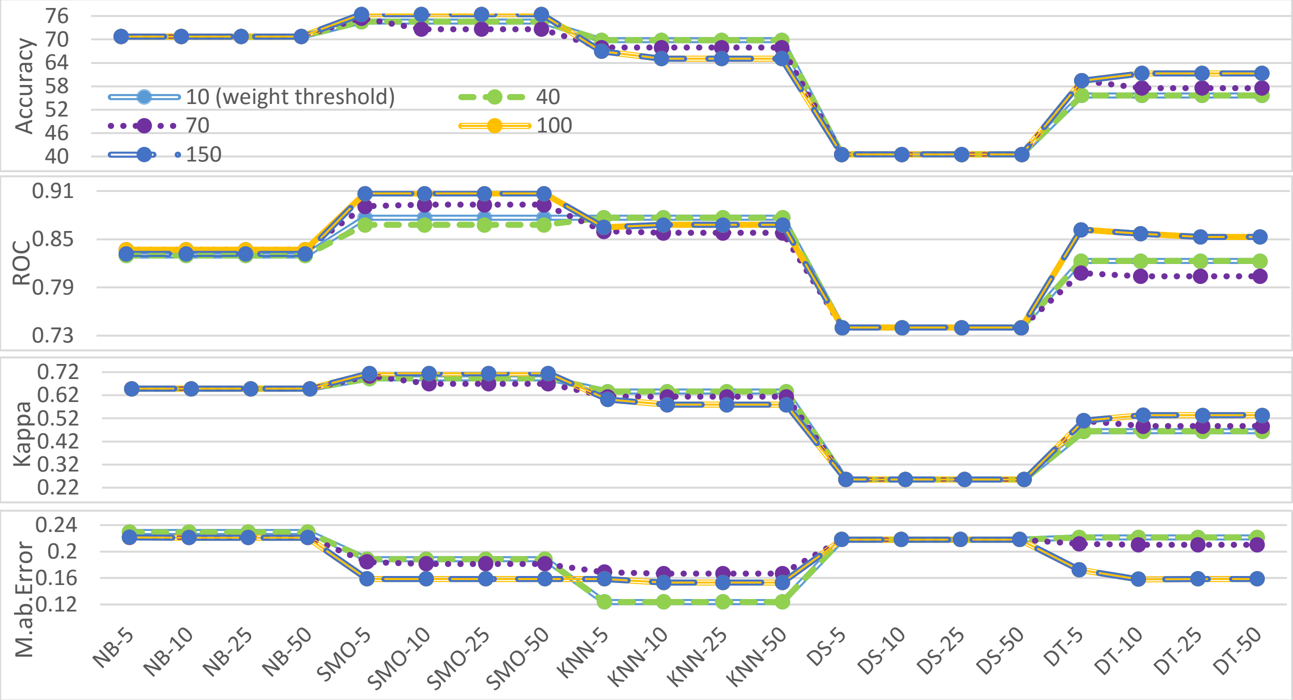


*Figure 4.7- The impact of selected parameters in AdaBoost algorithm*

NB-5: means that naive bayes classifier and 5 iteration were used

Considering the results using J48, the highest classification accuracy was obtained by using:

- Number of objects (instances) per leaf: 1
- Confidence factor of 0.5 and higher or unpruned

Figure4.8 shows the performance of J48 classifier with different configurations. The results show that the number of instances per leaf had the highest impact in the performance followed by the number of folds. The best accuracy was obtained by single instance per leaf and less pruning.

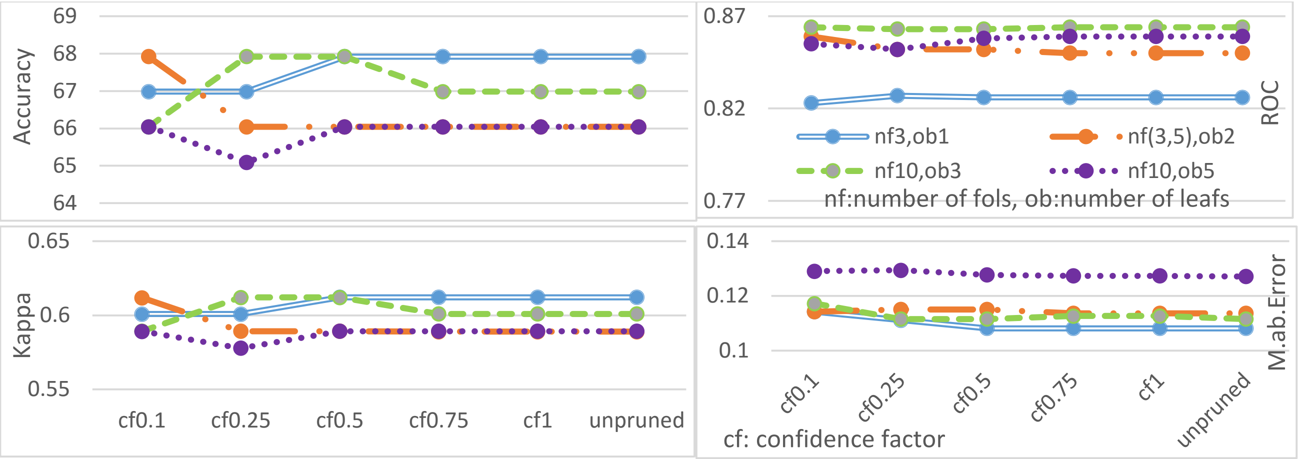


*Figure4.8- The impact of selected parameters in the J48 algorithm for Breast Tissue*

Considering the results using SVM, the highest classification accuracy was obtained by using:

- Polynomial kernel in SMO and linear kernel in SVM
- Standardization of data

Figure 4.9 shows the performance of the SMO and SVM classifiers. The result shows that SMO obtained in most cases better performance.

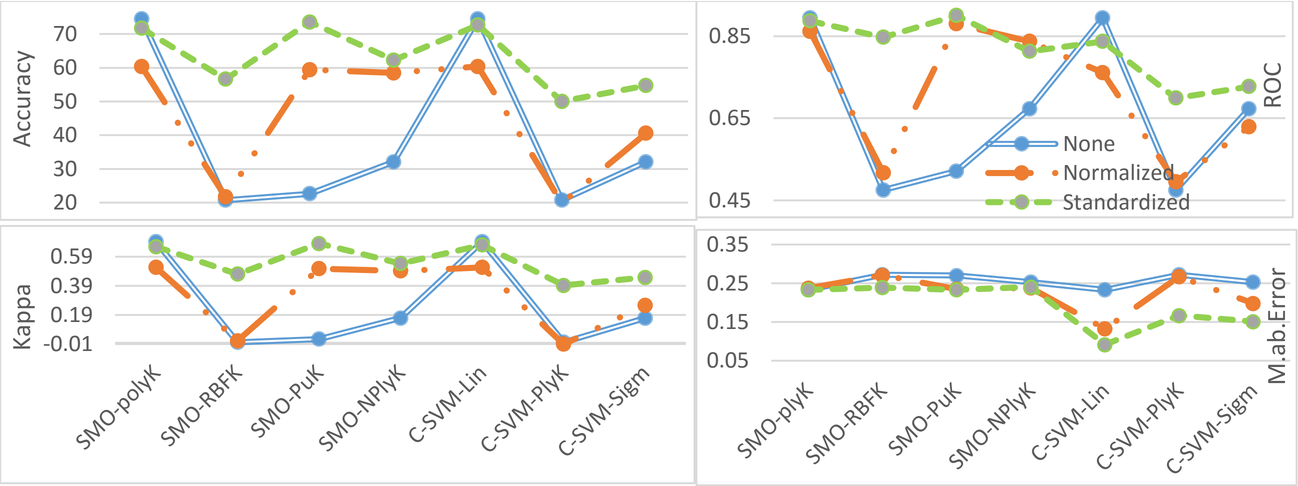


*Figure 4.9 - The impact of selected parameters in SVM & SMO algorithms*

We discretized features 7 and 9 of the dataset to four bins using Weka discretization tool. Figure 4.10 shows the results before and after the applying preprocessing. From Figure 4.10 we can see that the standardization was impacted on the performance of SMO and random forest. Discretization improved the results for AdaBoost.

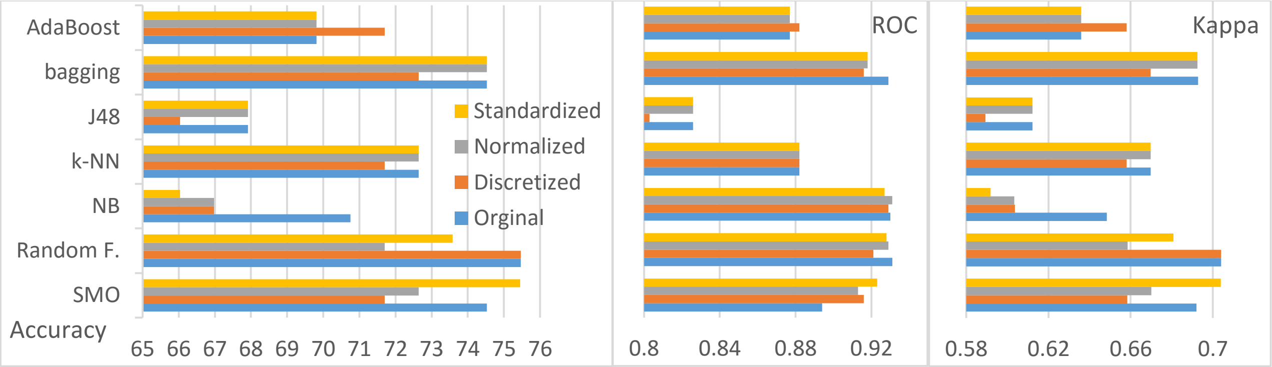


Figure 4.10- The impact of preprocessing methods in selected algorithms

## 4.2 Cardiotocography

For cardiotocography (CTGD), different classification techniques were applied by several other researchers. A multi-class classification algorithm using modular neural network applied by [Jadhav et al.2011] for classifying this data set. Random forest, linear discriminant analysis and reptree used by [Tomas et al.2013] to classify three-class labels version of dataset, resulting in 93% accuracy. A least squares support vector machine utilizing a binary decision tree was applied in [Yilmaz et al.2013]. With this algorithm authors could obtained accuracy of 91% at the three-class labels version. An adaptive neuro-fuzzy inference systems to predict two-classes on the cardiotocography dataset was proposed by[Ocak et al.2013]. A supervised artificial neural network was applied as classifier by [Sundar et al.2012]. SVM and genetic algorithms were evaluated in [Ocak 2013].

In this thesis we additionally applied ensemble techniques and meta-classifiers like bagging, cost-sensitive learning, and AdaBoost. Also the impact of parameters for chosen classifiers and different preprocessing techniques was evaluated. Figure 4.11shows the best performance achieved by selected algorithms (based on accuracy, kappa statistic, ROC and mean absolute error).

The best classification accuracy of 83.49% was achieved using random forest classifier. This was done by using max depth of each tree, with 100 trees and 40% of features. The second highest classification accuracy (76.11%) was obtained by the J48 classifier. This was done by using discretization of features with 6 bins (2-6,14,15,20) and an unpruned confidence factor of 0.25.

Considering the kappa statistics values, it is obvious from the results that it was difficult to predict one of the classes (the minor class). The random forest obtained the best classification accuracy.

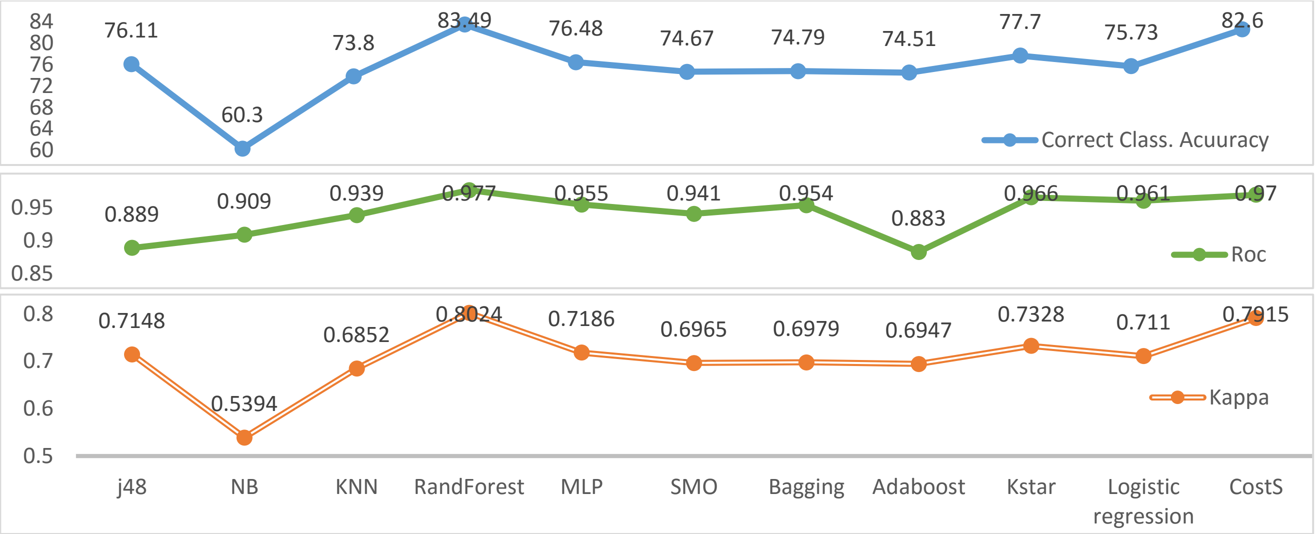


*Figure 4.11- The evaluation performance for CTGD using implementation of various classifiers in Weka*

With respect to the previous works on this dataset, authors tried to improve the performance by applying some feature selection techniques before starting the classification process. They also experimented with different preprocessing techniques and tried to tune classifiers by varying the configuration for significant parameters, which are presented and discussed briefly below.

Regarding the evaluation of k-NN algorithm, the highest classification accuracy was obtained by using:

- Weighting function (1/distance in Equation 2.4)
- k = 5

Figure 4.12 shows the performance of k-NN classifier for CTGD using different k values and different weighting functions. Results show that the classification accuracy decreases when the number of neighbors is increased. The best classification accuracy was obtained with the k value between 1 and 7. The mean absolute error also increased when k value was higher.

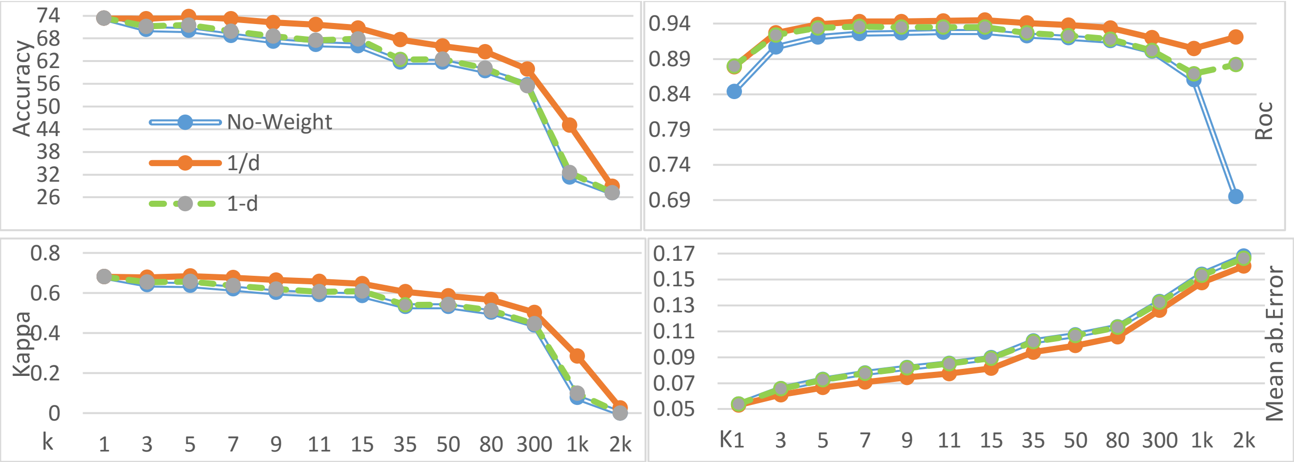


*Figure 4.12 - The impact of k and weighting function in k-NN algorithm*

When using random forest algorithm the highest classification accuracy was obtained by using:

- 40% of features
- 100 trees

- The minimum tree depth size of 18

Figure 4.13 shows the performance of this classifier using different number of features against a different number of trees and tree depth sizes. The results show that the classifier reached to a stable state after using 50 trees and a tree depth size of 18.

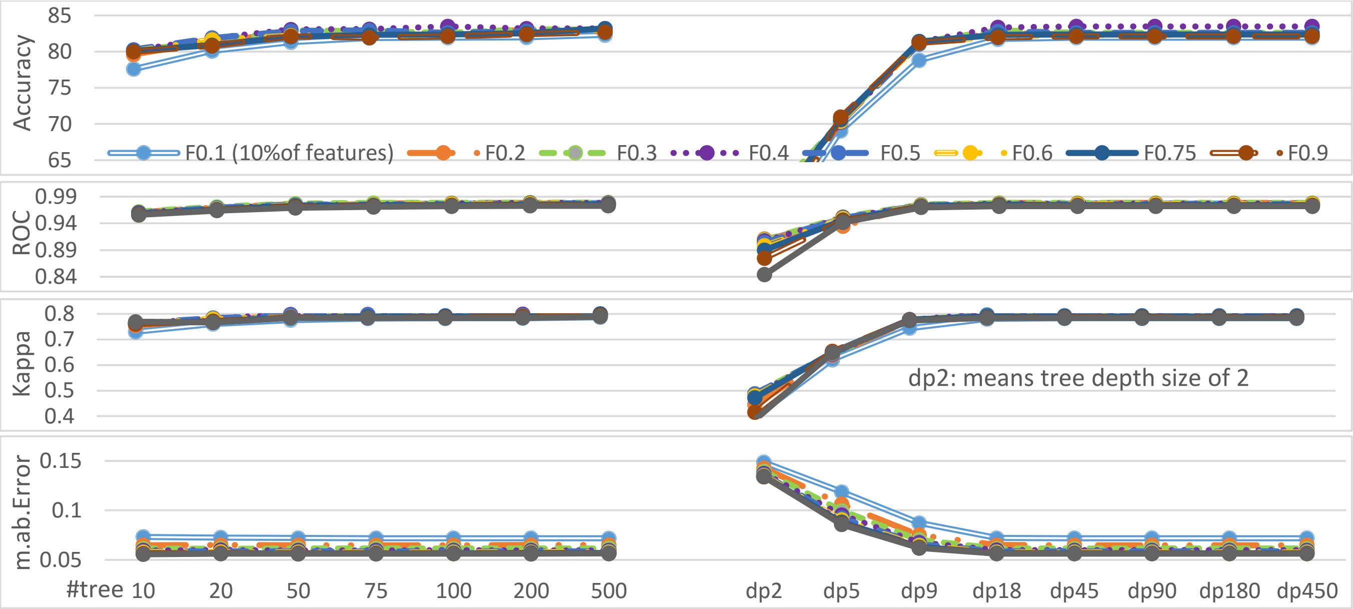


*Figure 4.13 - The impact of the number of features against the number of trees and the depth size of a tree in random forest*

Figure 4.14 shows the impact of the numbers of trees and tree depth size in the performance of random forest. It shows that the performance was improved when number of trees increased (visible improvement until 75 trees).

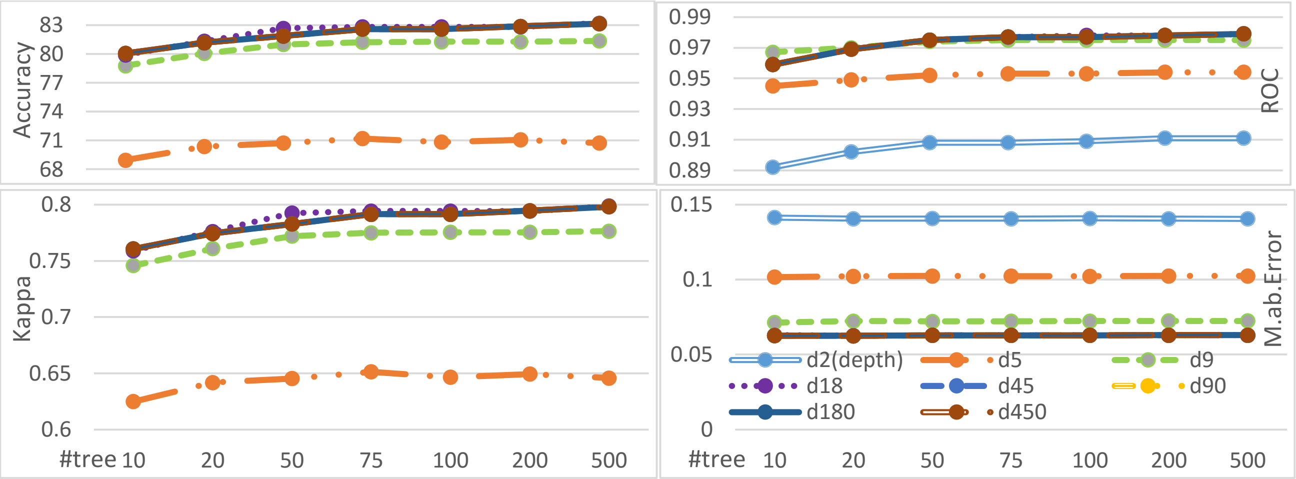


*Figure 4.14- The impact of the depth size of the tree against number of trees in the random forest algorithm*

Considering the results using bagging, the highest classification accuracy was obtained by using:

- SMO classifier
- Bag size of 40 and higher

Figure 4.15 presents the performance of this classifier. The results show that for small bag size of 10 a higher number of iterations improved the performance. It also shows that the bag size of 100 had a stable performance. The base classifier had the highest impact in the results.

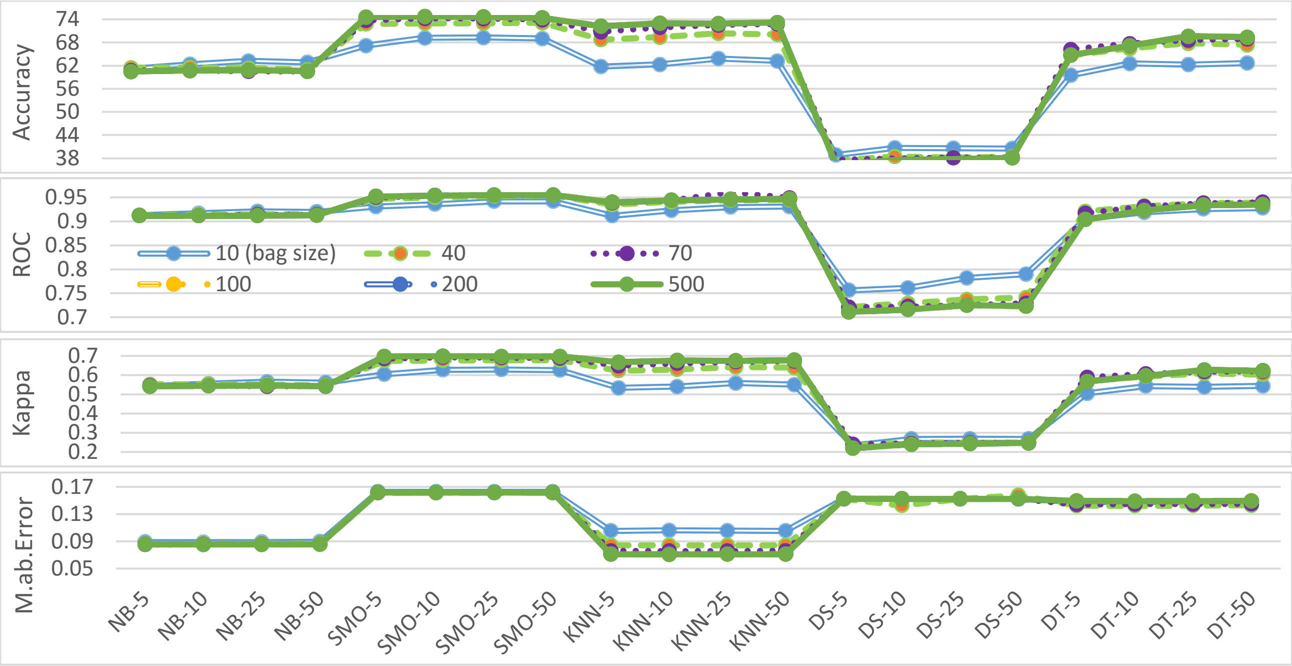


*Figure 4.15- The impact of selected parameters in the bagging algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using AdaBoost, the highest classification accuracy was obtained by using:

- SMO and the k-NN classifier
- Weight threshold 70 and higher
- 5 Iterations and higher

Figure 4.16 shows the performance of the classifier with different base classifiers, weight thresholds and iteration numbers. The results indicate that the base classifier had the highest impact of change in the AdaBoost performance. The best classification accuracy was obtained by using SMO, but this base classifier needed more execution time than other base classifiers.  The weight threshold had an impact only in the decision stump classifier.

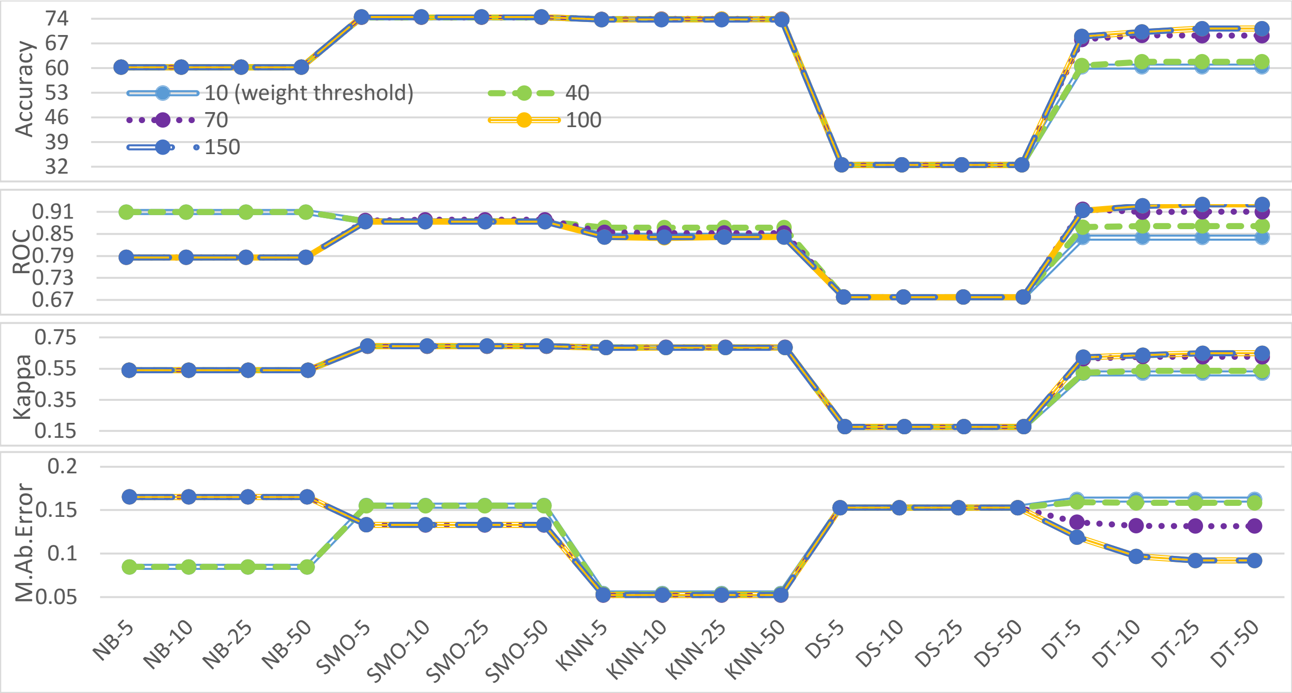


*Figure 4.16- The impact of selected parameters in AdaBoost algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using J48 classifier, the highest classification accuracy was obtained by using:

- Number of objects (instances) per leaf: 1
- Confidence factor of 0.75 and higher

Figure 4.17 shows the performance of J48 classifier with different classifier configurations. The results show that the number of instances per leaf had the highest impact in the performance followed by the number of folds. The best accuracy in this experiment was obtained by single instance per leaf. The change of the pruning parameter did not affect the results.

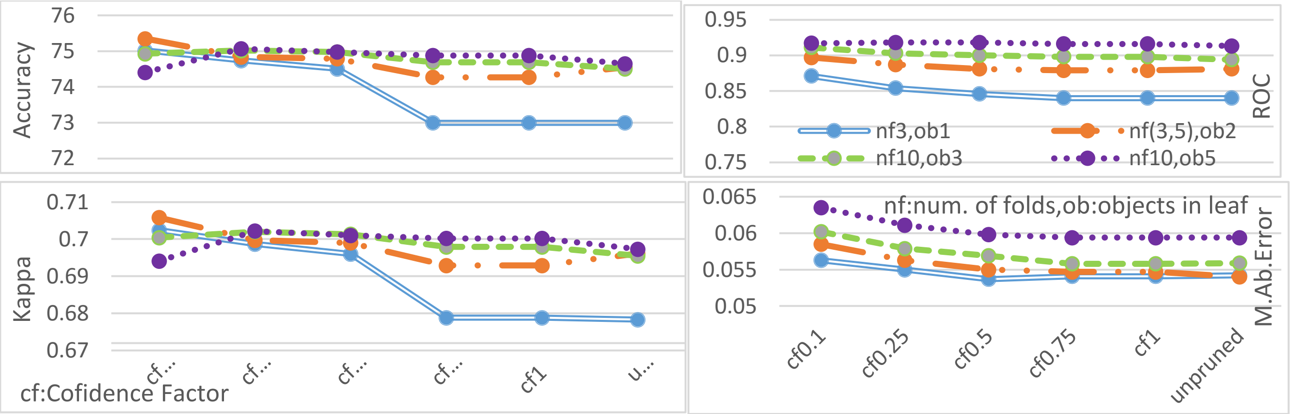


*Figure 4.17- The impact of selected parameters in the J48 algorithm*

Considering the results using SVM Classifier, the highest classification accuracy was obtained by using:

- RBF kernel in SMO and Polynomial kernel in SVM

- Without normalization and standardization of data

Figure 4.18 shows the performance of the SMO and SVM classifiers. The results indicate that the SMO classifier obtained better performance in most cases.

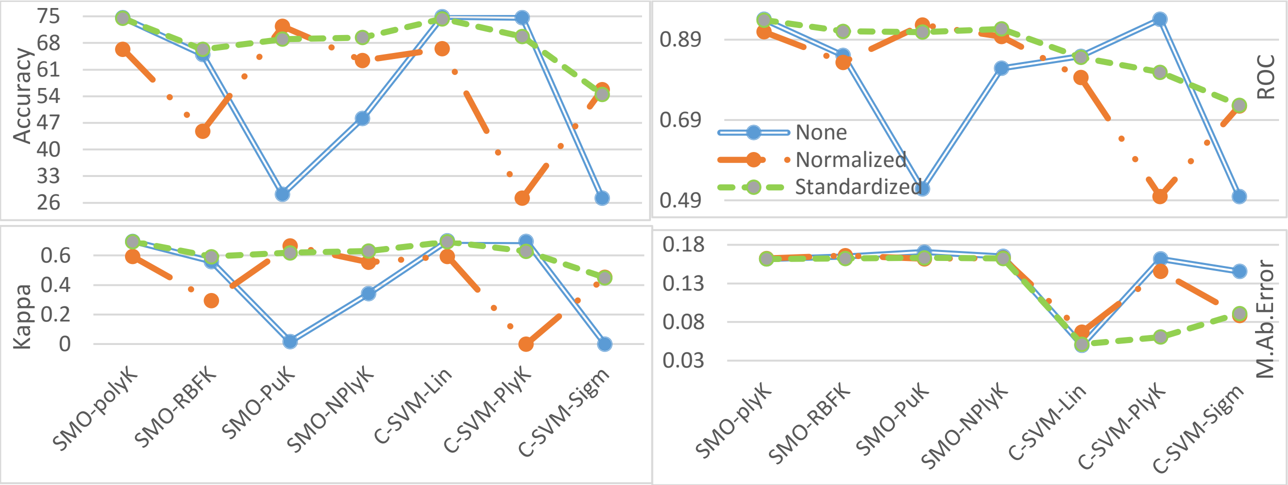


*Figure 4.18- The impact of selected parameters in SVM & SMO algorithms*

We discretized features 7 and 9 of the dataset to four bins using Weka discretization tool. Figure 4.19 shows the results before and after applying the preprocessing on selected classifiers. It indicates that the discretization had impact on the performance of SMO and J48.

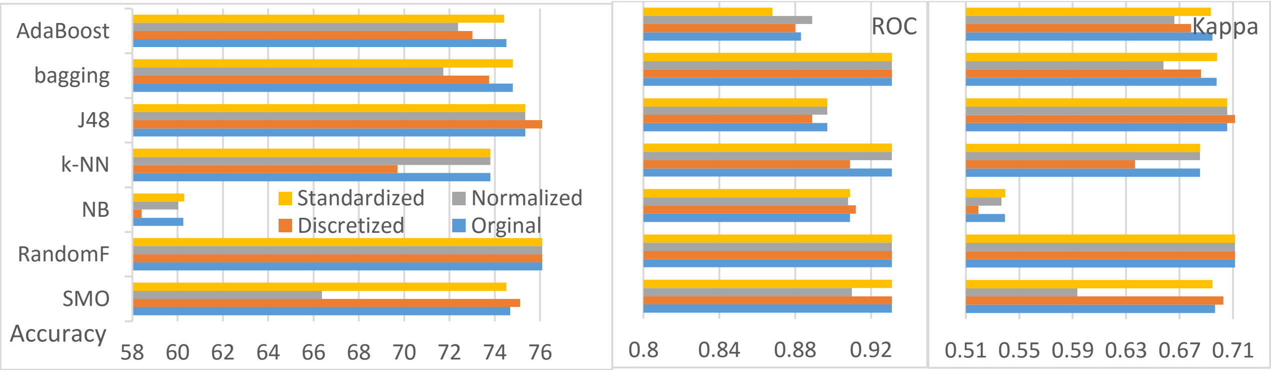


*Figure 4.19- The impact of the preprocessing methods in selected algorithms*

## 4.3 Diabetic Retinopathy

For diabetic retinopathy (DRDD) different techniques have been used. Ensemble learning techniques were applied in [Antal et al. 2014] which called an ensemble-based system for automatic screening of diabetic retinopathy and achieved the best classification accuracy of 90%. In [Mane et al.2014] authors reviewed the used machine learning techniques on this dataset. A Kernel extreme learning technique was applied by [YiNan et al.2016]. Deep neural networks were applied in [Haloi 2015] with a classification accuracy of 96%.

Based on the papers that were found for this dataset, the best results were obtained by [Haloi 2015] using a deep neural network with classification accuracy of 96% followed by [Antal et al. 2014], which used ensemble learning techniques with a classification accuracy of 90%.

In this thesis we additionally applied k-star, and other ensemble techniques. Figure 4.20 shows the best achieved results by selected classifiers. AdaBoost obtained the best classification accuracy of 74.8%. This was done by using SMO as the base classifier with the help of polynomial kernel. The second highest classification accuracy was obtained by logistic regression with an accuracy of 74.7%.

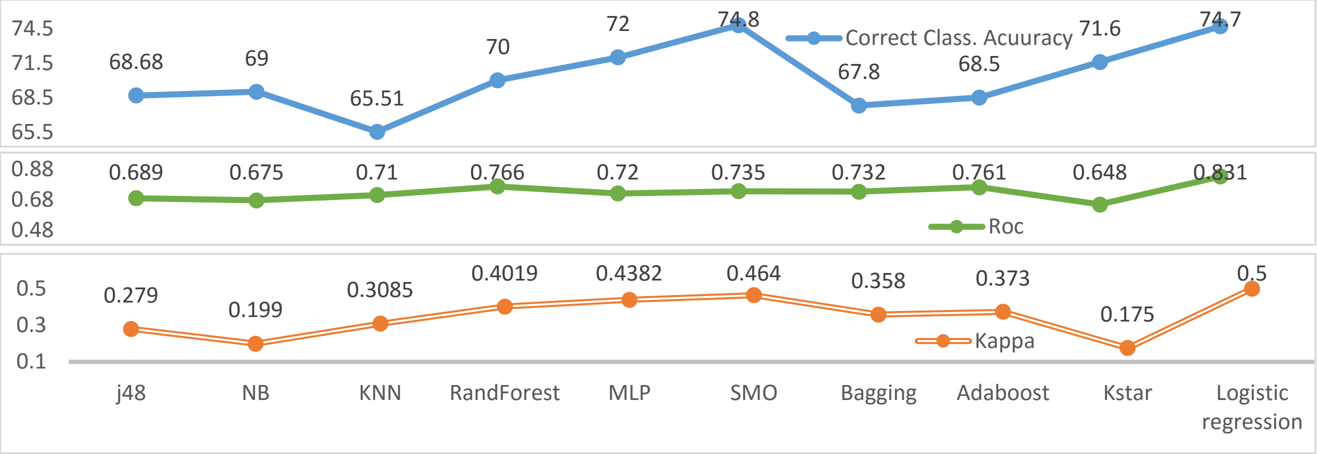


*Figure 4.20- The evaluation performance on DRDD using implementation of various classifiers in Weka*

With respect to the previous works on this dataset, authors tried to improve the performance by applying some feature selection techniques before starting the classification process. We experimented this dataset with different preprocessing techniques and tried to tune classifiers by varying the configurations, which are presented and discussed briefly below:

Regarding the evaluation of k-NN classifier, the highest classification accuracy was obtained by using:

- Weighting function: 1/distance (see Equation 2.4)
- k = 21

Figure 4.21 shows the performance of k-NN classifier for DRDD using different k values against a different weighting functions. The classification accuracy was improved, by raising the k value. it had even an acceptable result till k=100 with the weighting function. But in the opposite way, the value of mean absolute error increased because the classifier tends to classifier one of the class labels better (in this case by using weighting function, following confusion matrixes (CM) obtained for k=100 CM={438,102;291,320} and for k=35 CM={497,143;254,357}. Although we had least correct classification on k=100 (757 vs 854 in k=35) the classification accuracy is 0.3% is higher because miss classification ratio is move from class label1 to class labl2).

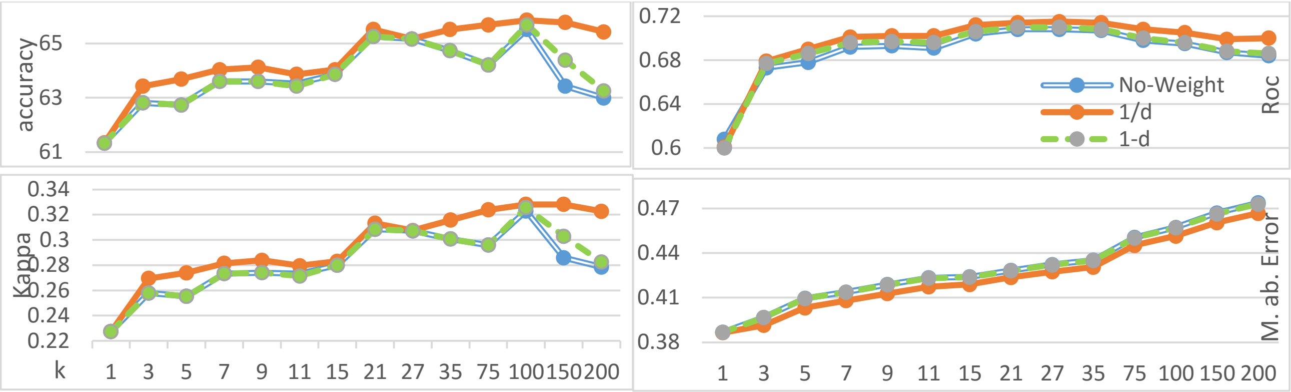


*Figure 4.21 - The impact of k and weighting function in k-NN algorithm*

Considering the results using random forest, the highest classification accuracy was obtained by using:

- 90% of features
- 75 trees
- Minimum tree depth size of 18 and higher

Figure 4.22 shows the performance of this classifier using a different number of features against a different number of trees, and the depth size of a tree. The results show that the classifier reached a stable state after using a tree depth size of 18.

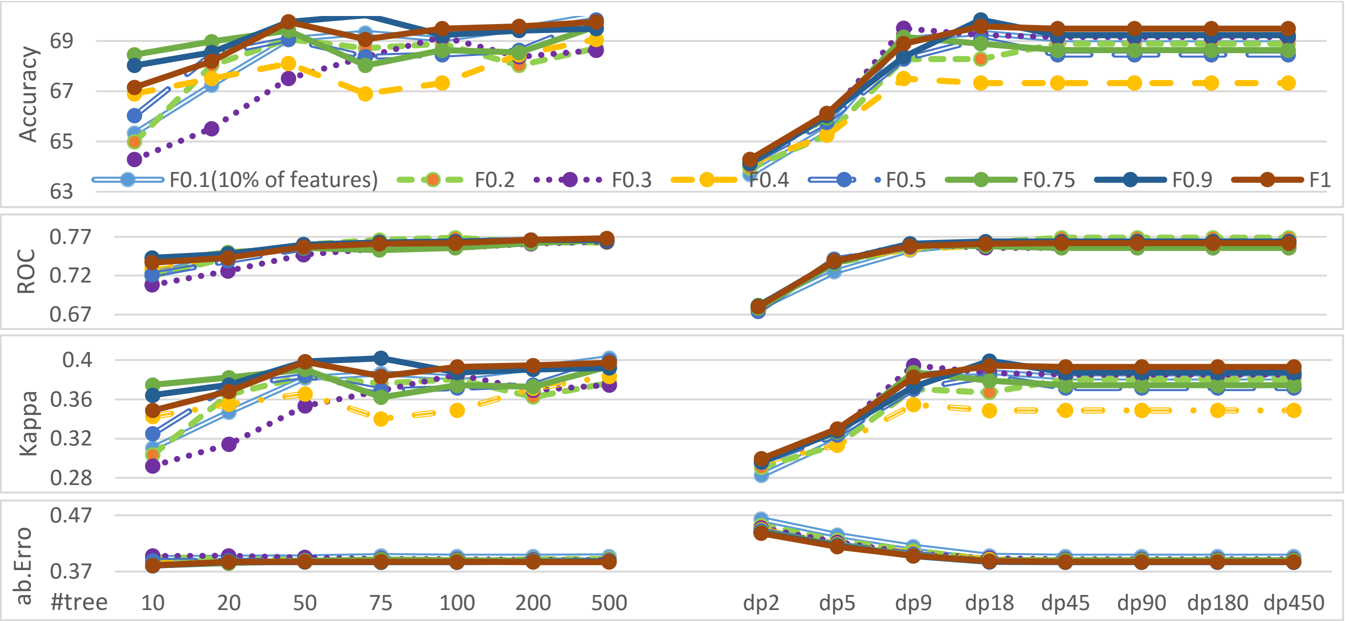


*Figure 4.22 - The impact of the number of features against the number of trees and the depth size of a tree in random forest*

Figure 4.23 shows the impact of the number of trees and the depth size of the trees. The best performance was obtained by using the tree depth size of 18.

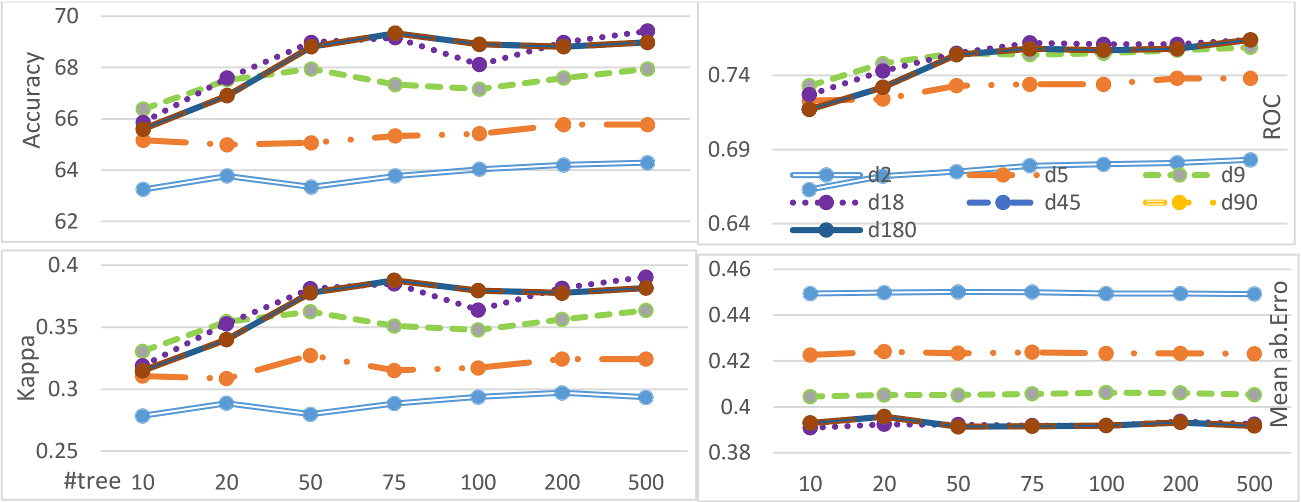


*Figure 4.23- The impact of the depth size of tree against number of trees in the random forest*

Considering the results using bagging, the highest classification accuracy was obtained by using:

- SMO classifier
- Bag size of 70 and higher
- 10 iterations

Figure 4.24 shows the performance of this classifier with different base classifiers, number of iterations and bag sizes. Regarding base classifier, the best results were obtained by SMO.

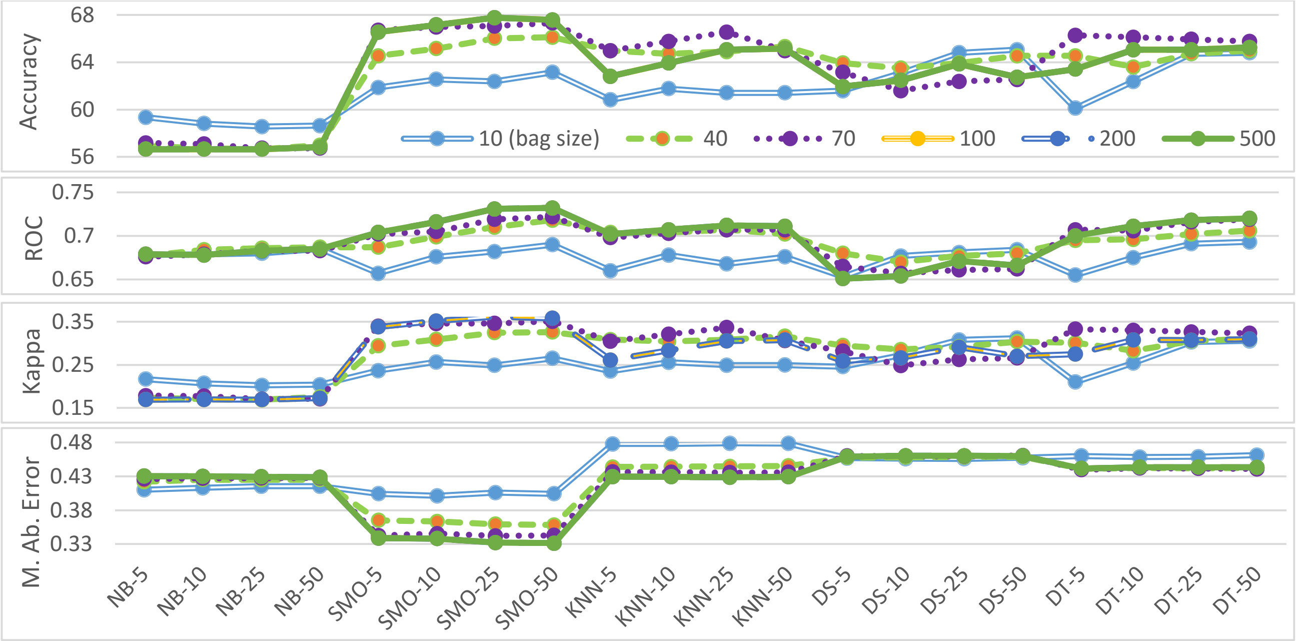


*Figure 4.24- The impact of selected parameters in the Bagging algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the evaluation of AdaBoost, the highest classification accuracy was obtained by using:

- SMO classifier
- Weight threshold 100 and higher
- 25 Iterations

Figure 4.25 shows the performance of the classifier with different base classifiers, iteration numbers and weight thresholds. The result indicates that the base classifier had the maximum impact in the performance. In this experiment the best accuracy was obtained by using the SMO, however it this algorithm was required more time.

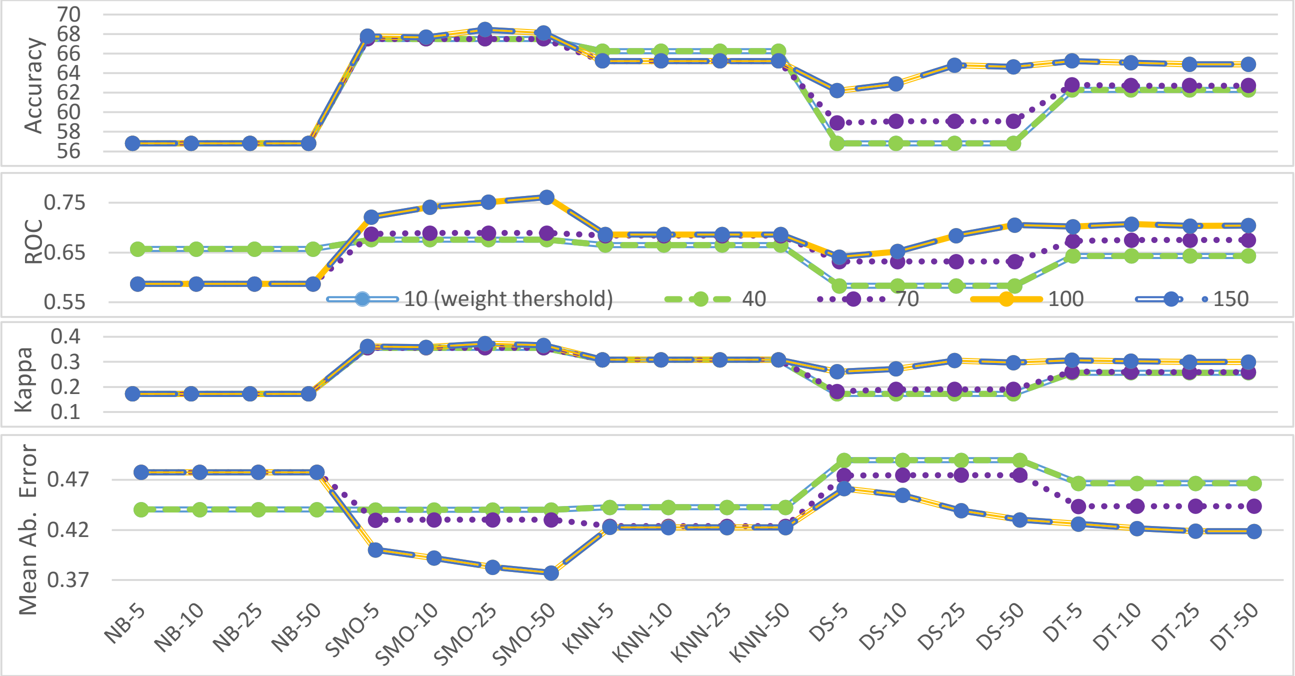


*Figure 4.25- The impact of selected parameters in AdaBoost algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using J48, the highest classification accuracy was obtained by using:

- Number of objects (instances) per leaf: 2
- Confidence factor: 0.1

Figure 4.26 shows the performance of the classifier with different parameters configurations. The results indicate that the number of instances per leaf had the highest impact on the performance followed by the number of folds. In the experiment the best accuracy was obtained by two instances per leaf.

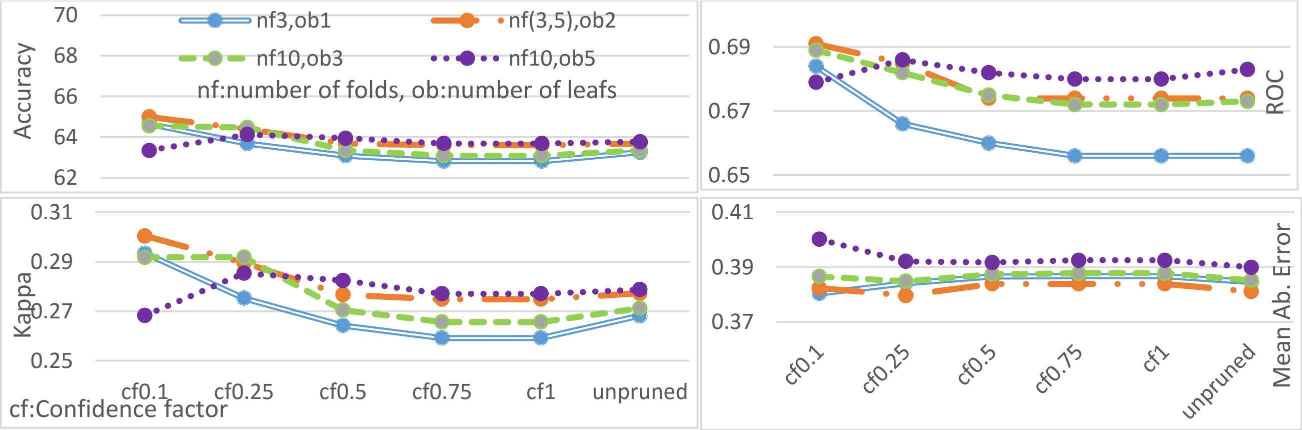


*Figure 4.26- The impact of selected parameters in the J48 algorithm*

Considering the results using SVM, the highest classification accuracy was obtained by using:

- Polynomial kernel in SMO and Linear kernel in SVM
- With original and followed by standardization of data

Figure 4.27 shows the performance of SVM and SMO classifiers. The experiment resulted in a higher accuracy with original datasets in both classifiers. The results also show that SMO obtained in most cases better performance.

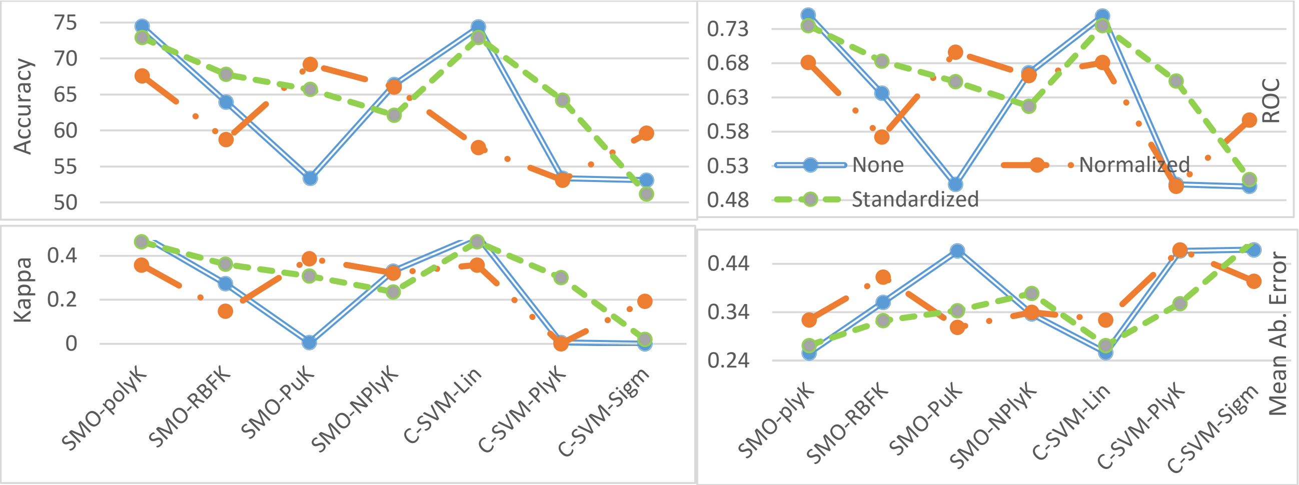


*Figure 4.27 - The impact of selected parameters in SVM & SMO algorithms*

Figure 4.28 shows the performance before and after applying preprocessing. We discretized features 1, 2 and 19 using two bins and features 12 to 17 using seven bins in the Weka discretization tool. The preprocessing techniques improved the classification accuracy of the classifiers up to 6 %. The result also shows that the standardization improved the results in both bagging and AdaBosst.

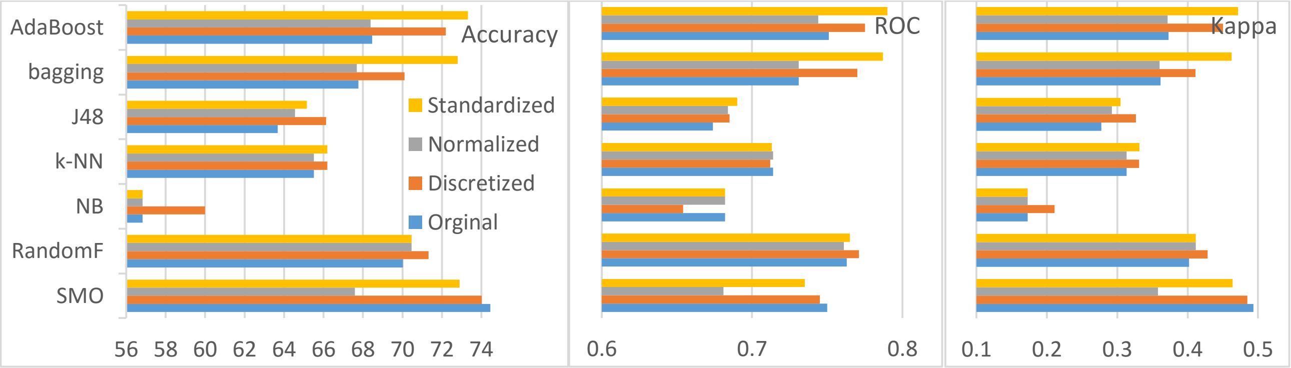


*Figure 4.28- The impact of preprocessing methods in the selected algorithms*

## 4.4 Fertility

Different classification techniques were used for fertility dataset (FD). Decision trees, multilayer perceptron (MLP) and support vector machine (SVM) were applied in [Gil Mendez et al.2012] with the highest prediction accuracy values of 86% which was obtained by both MLP and SVM. To be noted is, that the provided UCI dataset had a tiny difference from the dataset used in the paper. The classification was performed considering three different values for the class decisions: sperm concentration, the percentage of motile sperm or normal morphology. A proposed method using a nearest-prototype-style classifier was used in [Chalmers et al.2014] which was optimized with a genetic algorithm. A supervised ensemble learning method called clustering-based decision forests was proposed in [Wang et al. 2014] as an ensemble classifier. The authors compared their classifier performance with CART, SVM, multiplayer perceptron and logistic regression. The area under the ROC was used for measuring the performance. The authors obtained the mean of AUC (ROC) with a value of 0.916. The bayesian belief network classifier with the search algorithm hill climbing method was applied in [Naeem, 2014] to classify the dataset, resulting in 91% classification accuracy. The authors in [Zhang et al. 2015] compared SVM, naive bayes and J48 classification methods using different performance measurements including classification accuracy, sensitive, specificity, precision, recall, G-means and F-measure. They investigated the differences between classification accuracy in imbalanced/balanced datasets using six different datasets. The genetic programming was applied in [Dufourq et al. 2013]. In this work also the arithmetic tree, logical trees, and decision trees were used as three different representations for classification methods, resulting in 82%, 84.8%, and 80% classification accuracy. The logical tree method achieved the best classification accuracy of 84.8% for this dataset. The classification algorithms Cart and C4.5 were applied in [H., 2012], respectively with 55% and 87% classification accuracy. The best classification accuracy of 91% was obtained by [Naeem, 2014] using Bayesian Belief Network.

In this thesis we additionally applied random forest, logistic regression, reptree and meta-classifiers (ensemble techniques) including bagging, cost-sensitive learning classifier and vote-classifier. Different preprocessing techniques and various parameters configuration were applied on those classifiers.

The best results of the used classifiers are shown in Figure 4.29. Regarding classification accuracy, the best classification accuracy of 90% was achieved by using AdaBoost. This was done by applying discretization for the preprocessing phase on features 2, 7 and 9. The k-NN was used as the base classifier with k=5. The second highest classification accuracy of 90% was obtained by bagging, J48 and MLP.

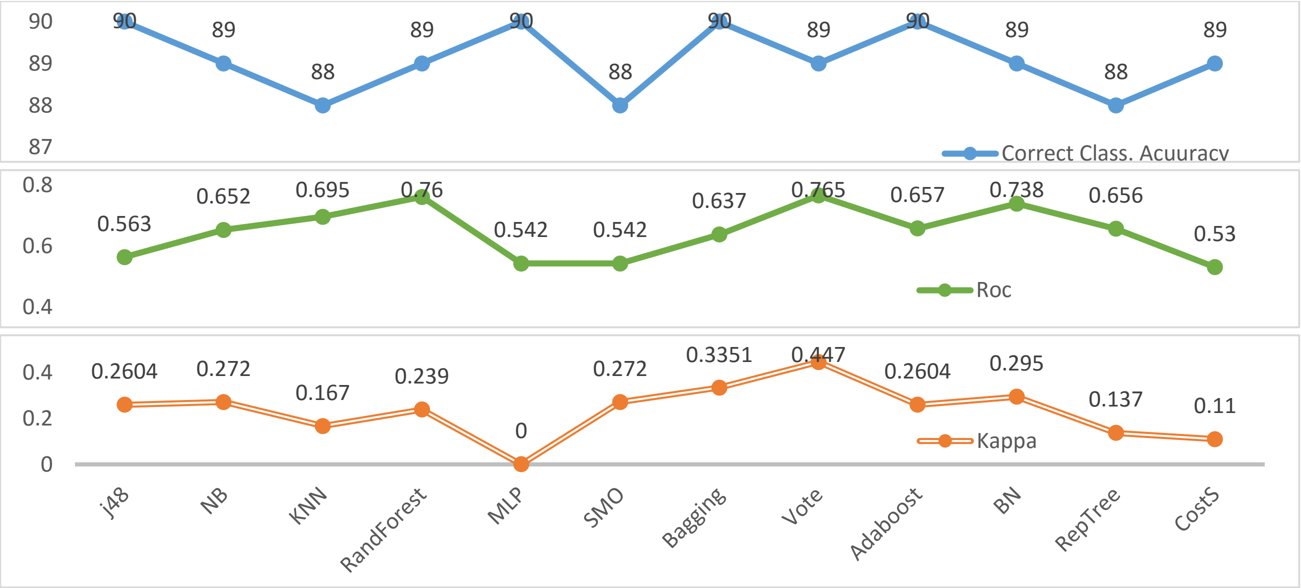


*Figure 4.29- The evaluation performance of the Fertility dataset using implementation of various classifiers in Weka*

With respect to the previous works on this dataset, authors tried to improve the performance by applying some feature selection techniques before starting the classification process. In our experiments we achieved higher classification accuracy than [Gil Mendez et al.2012] and [Zhang et al. 2015] by using MLP and SMO classifier. We also experimented with different preprocessing techniques and tried to tune classifiers by varying the configuration for significant parameters, which are presented and discussed briefly below.

Regarding the evaluation of k-NN classifier, the highest classification accuracy was obtained by using k = 1. Figure 4.30 shows the results of k-NN classifier using different k values and weighting functions.

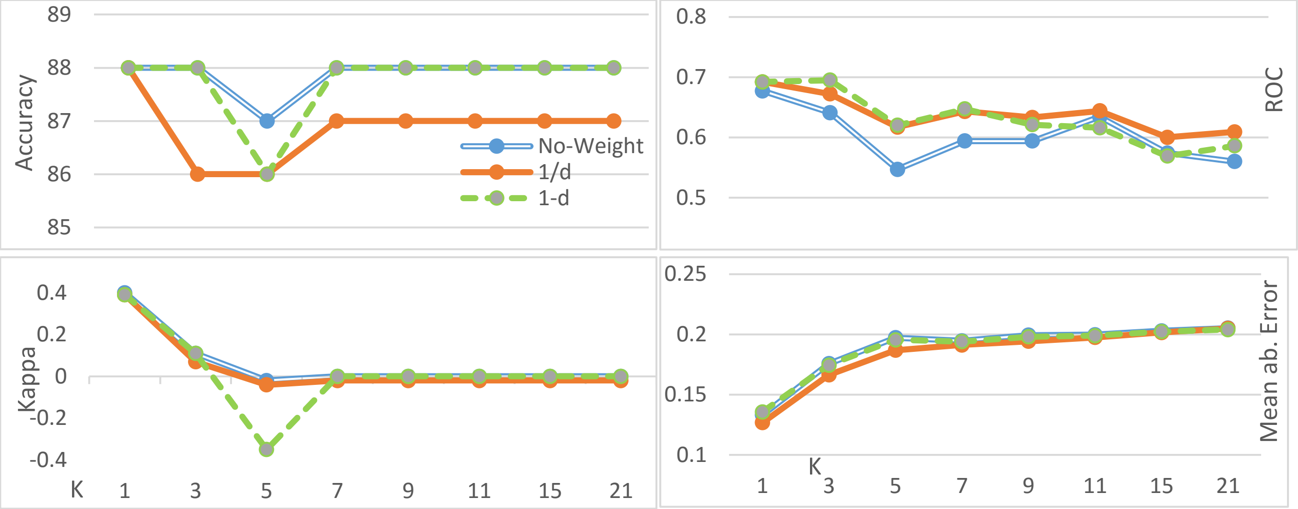


*Figure 4.30- The impact of k and weighting function in k-NN algorithm*

Considering the evaluation of random forest, the highest classification accuracy was obtained by using:

- 10% and 50% of features
- 10 and 20 number of trees
- The tree depth size of 5

Figure 4.31 shows the performance of this classifier using different number of features and different number of trees. The high number of features should be used if using a lower number of trees. Using the 50% of features resulted in much better performance in the classifiers.

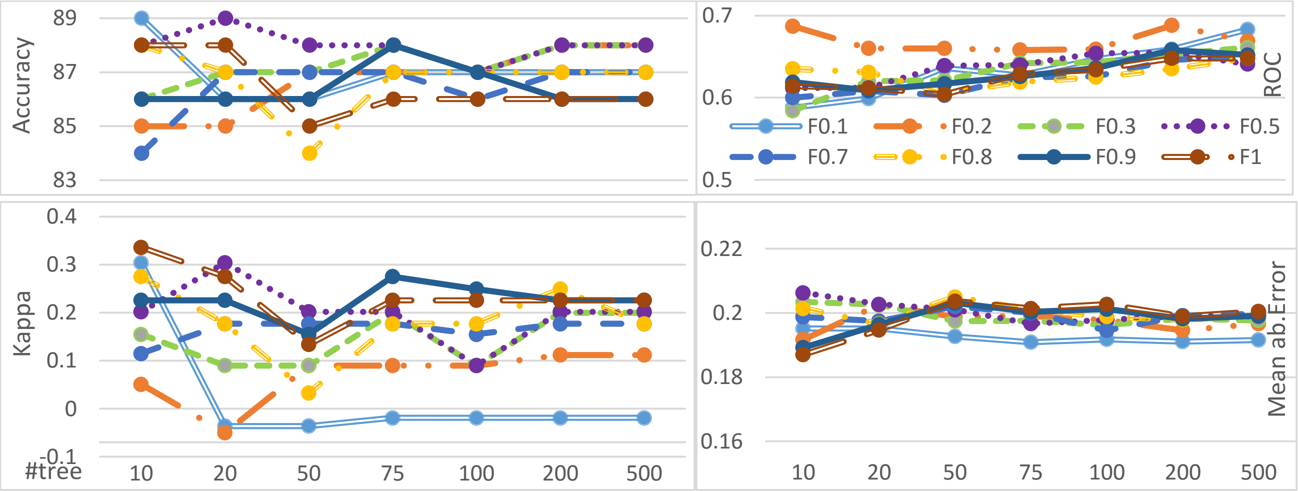


*Figure 4.31 - The impact of the number of features and the number of trees in the random forest*

Figure 4.32 shows the results of the algorithms using different number of trees and depth sizes. It indicates that the higher number of trees improved the performance and the depth size of 9 and higher achieved better performance (higher kappa values and least mean absolute error).

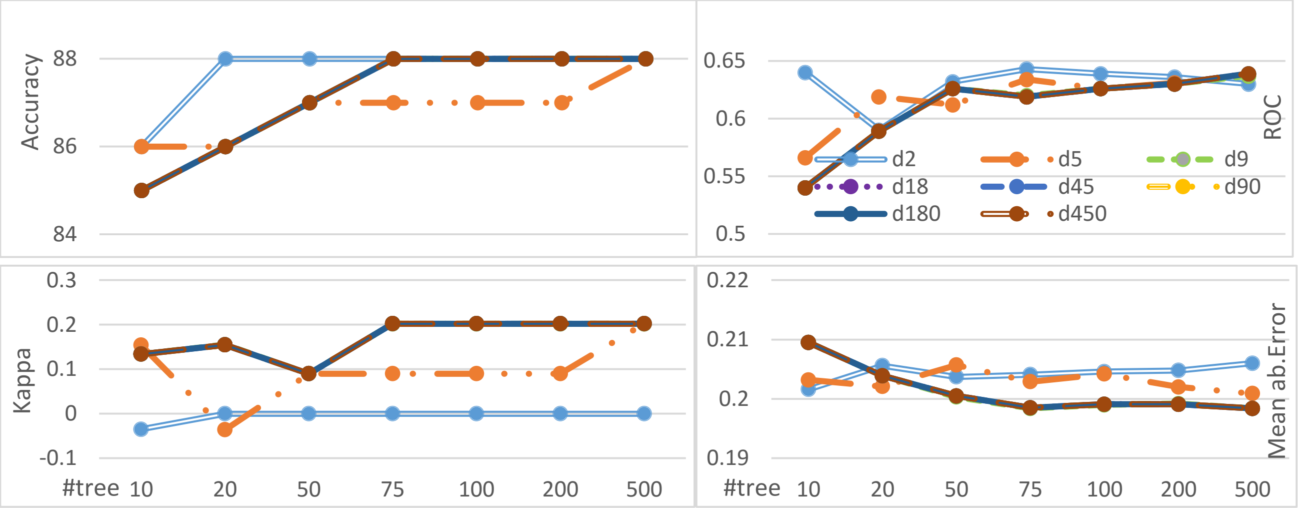


*Figure 4.32 – The impact of the depth size of tree and number of trees in the random forest*

Figure 4.33 shows the results of evaluation using different numbers of features and tree depth sizes. The results indicate that the depth size of 5 obtained the best accuracy rate.

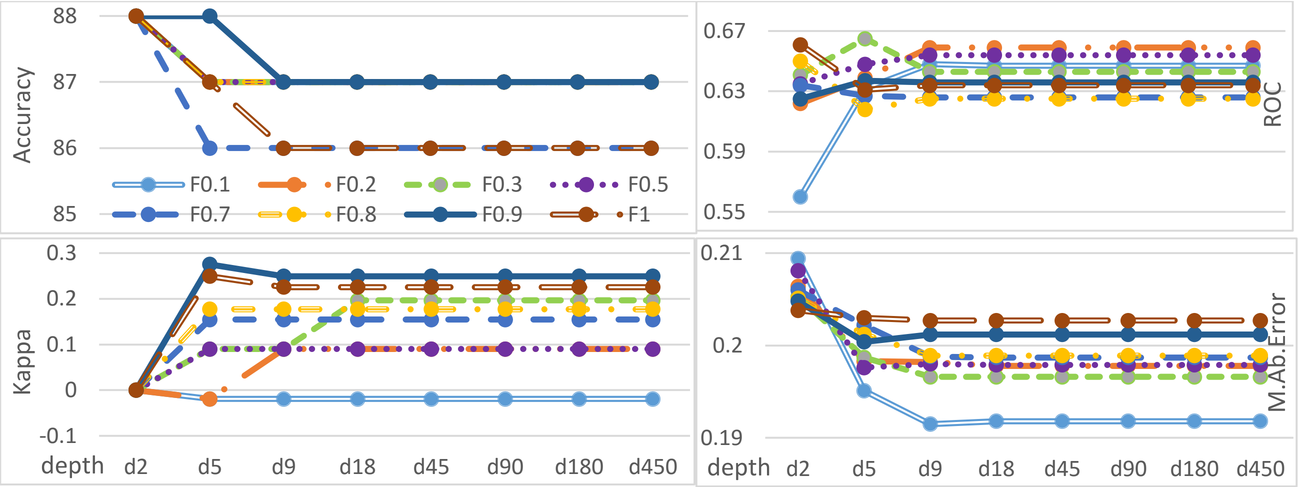


*Figure 4.33 - The impact of the number of features and the depth size of tree in random forest classifier*

Considering the results using bagging, the highest classification accuracy was obtained by using:

- Bag size 10 and 500
- 5 and 50 iterations

Figure 4.34 shows the results of bagging with different parameter configurations. The results indicate that the differences in the performance were not significant. We can see that k-NN had the best performance regarding kappa statistic results.

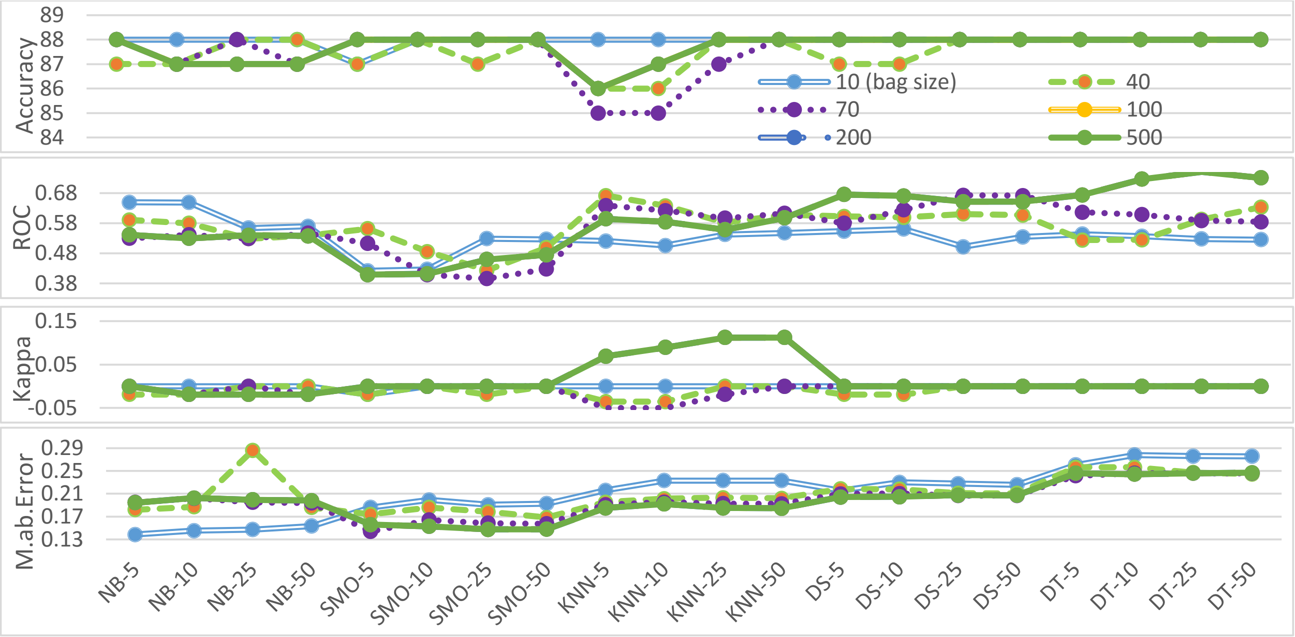


*Figure 4.34- The impact of selected parameters in the bagging algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the evaluation of AdaBoost, the highest classification accuracy was obtained by using:

- Weight threshold 10 and 40
- 5 Iterations and higher

Figure 4.35 shows the performance of the classifier with different base classifiers, iteration numbers and weight thresholds. The weight threshold had the maximum influence on the performance. The best accuracy was obtained by using weight threshold of 10 and 40.

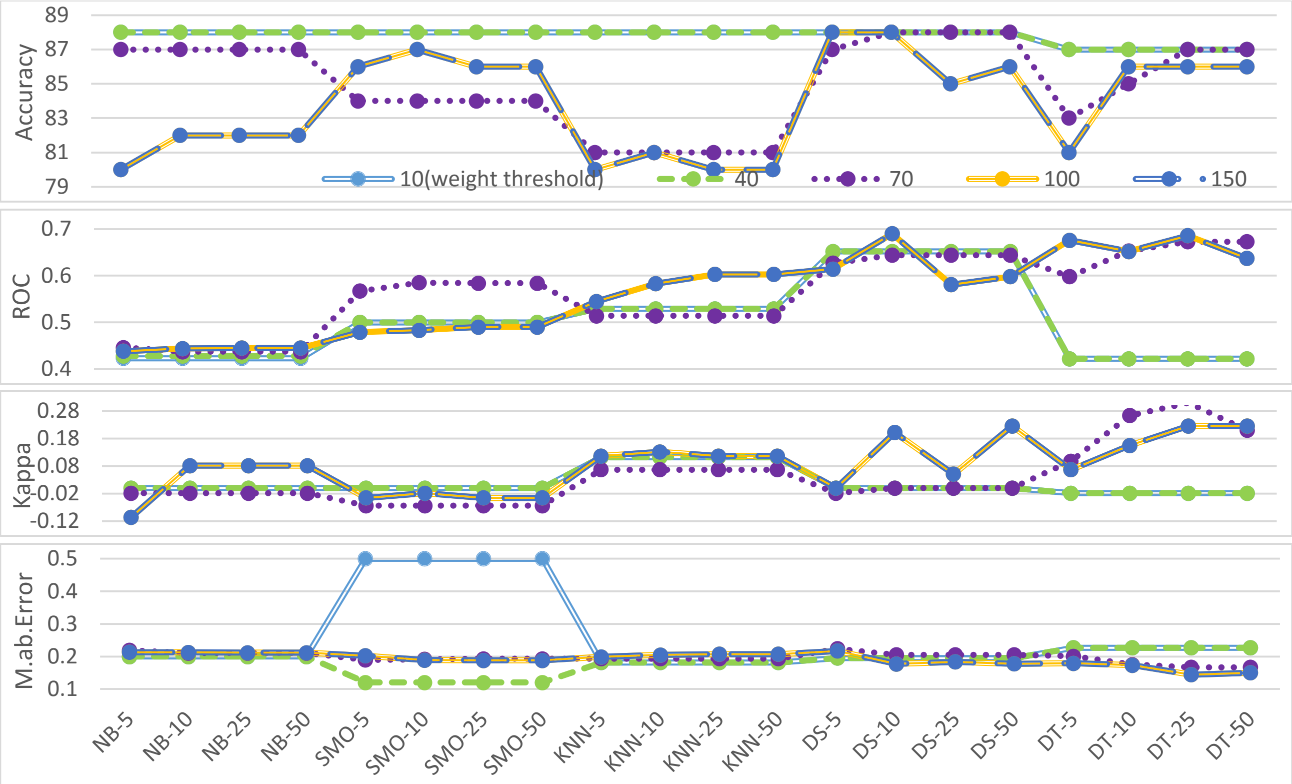


*Figure 4.35- The impact of selected parameters in AdaBoost algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using J48, the highest classification accuracy was obtained by using Confidence factor of 0.1.

Figure 4.36 shows the performance of J48 classifier with using different number of objects per leaf, number of folds and confidence factors. The results indicate that pruning had the maximum effect on the performance.

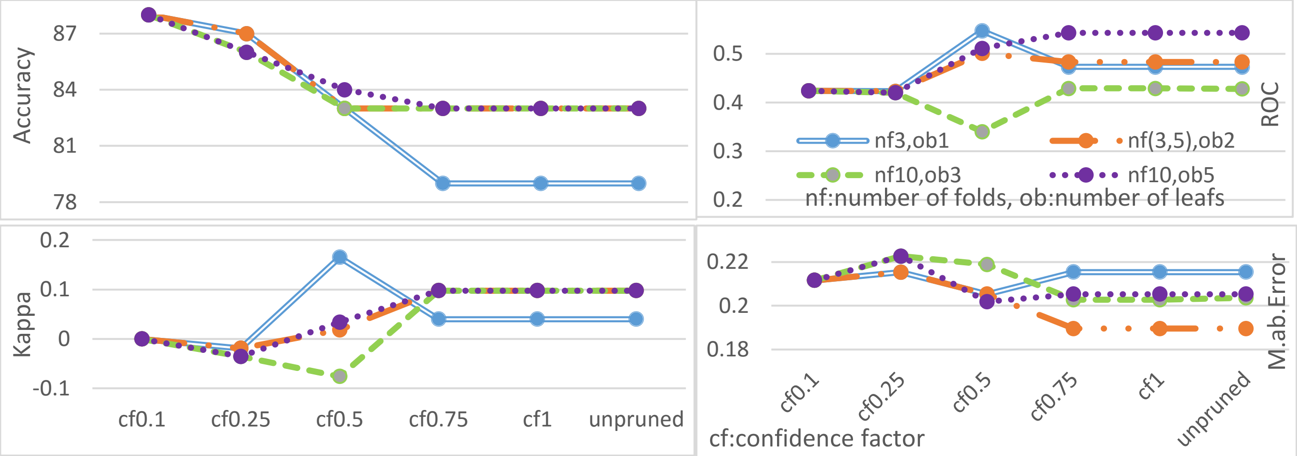


*Figure 4.36 - The impact of selected parameters in the J48 algorithm*

Considering the evaluation of SVM, the highest classification accuracy was obtained by using:

- RBF kernel in SMO and all kernel in SVM
- Without normalization or standardization of data

Figure 4.37 shows the performance of SMO and SVM classifier.

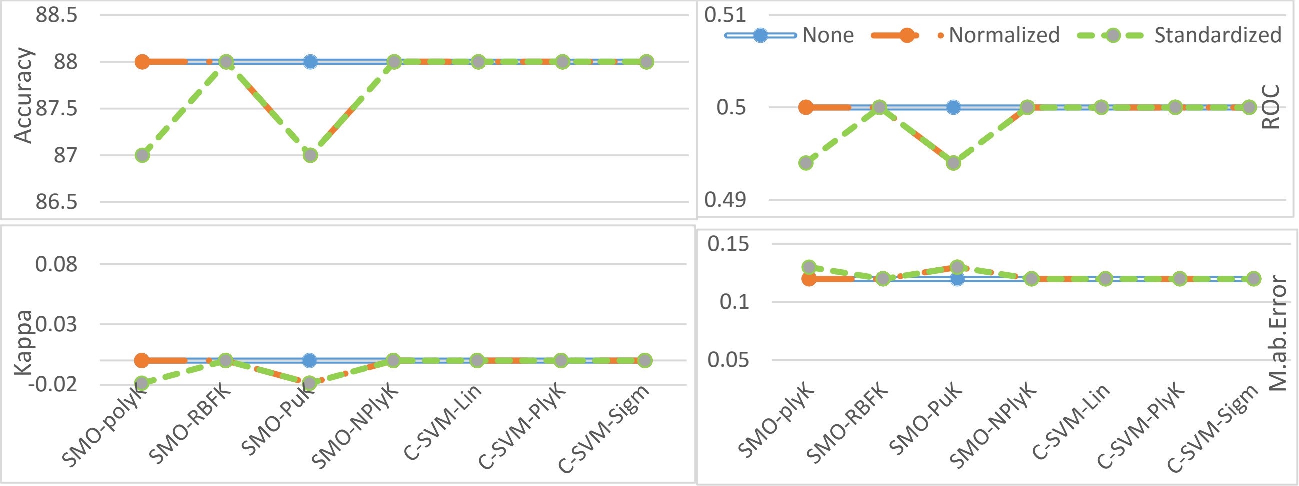


*Figure 4.37- The impact of selected parameters in SVM & SMO algorithms*

Figure 4.38 shows the results of performance before and after applying the preprocessing. The features 2, 7 and 9 of this dataset were discretized using four bins in Weka discretization tool. The discretization improved the accuracy of the tested classifiers till 2 %. The results also show that the normalization also improved the results in bagging by using base classifier of k-NN.

The discretization of “age”, “freq_alcohol consume” and “sitting hours” improved the classification accuracy in Bayes networks, k-NN, NB, MLP, SMO.

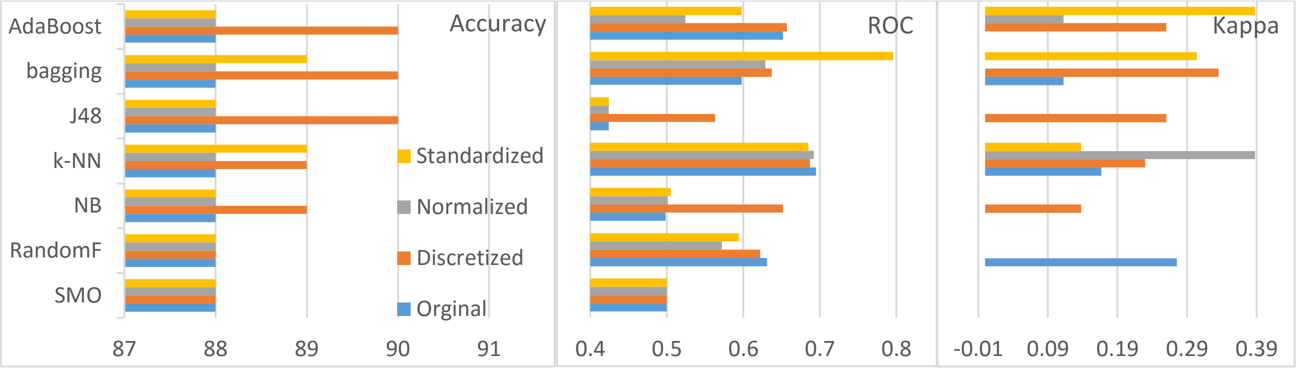


*Figure 4.38- The impact of preprocessing methods in the selected algorithms*

## 4.5 Indian Liver

For Indian Liver Patient dataset (ILPD) different techniques have been used. k-nearest neighbor classifier was applied in [Ramana et al.2012], when the authors tried to find out why this classifier performs better in this dataset in comparison with other dataset from USA patients. In [Ramana et al.2012_2], a modified rotation forest algorithm has been used. The baseline classier in the forest was chosen from a set of classifiers including tree-, statistical-, neural networks-, rule- and lazy- based learners. J48, MLP, random forest, linear regression, SVM and genetic programming were applied in [Jankisharan et al. 2014]. The best classification accuracy of 84% was obtained using original dataset by genetic programming. Authors additionally tried to balance the dataset, by applying the random forest method achieved 89% classification accuracy. The hierarchal clustering algorithms have been applied in [Babu et al. 2014] for classification and achieved a classification accuracy of 80%. Logistic, linear logistic regression, bayesian logistic regression, logistic model trees, multilayer perceptron, k-star, ripper, neural networks, rule induction, SVM and CART were applied in [Bahramirad et al. 2013]. The authors achieved a high classification accuracy of 97.33% by using Bayesian boosting as a classifier with the help of “optimization on rule induction method”. In [H. & M. 2014] authors applied SVM and evaluated this classifier by using different feature selection strategies, which lead to 73.2% classification accuracy while using 8 of 10 features. The decision tree learner (ID3) was applied in [Camilleri et al.2014] considering simple genetic algorithms as meta-optimizer for finding best parameters. The authors obtained 100% accuracy by using the percentage-split evaluation technique (70% Learning / 30% Test). The back propagations learning (BP), radial basis function network (RBF), self-organizing map (SOM) and SVM classifier techniques were utilized in [Tiwari et al. 2013]. The authors split the dataset into two parts considering genders and achieved a high classification accuracy of 98% by SVM. Cart and C4.5 were applied in [H., 2012]. These algorithms achieved a classification accuracy of 64% and 70%, respectively. CART was applied also in [Hyontai 2013]. The authors obtained a classification accuracy of 85.8% for the same dataset. Naïve Bayes and SVM have been performed in [V. & D. 2015]. The best accuracy of 79.6% was achieved by SVM classifier. In [Liang et al. 2013] authors proposed a combination of artificial immune system and genetic algorithm to classify the dataset. This method obtained a very high classification accuracy of 98.1 %. The extreme learning machine (ELM) has been applied in [Ertugrul et al.2014] using the following novel approaches: single layer ELM, tuning ELM and ELM based on linear regression. With this algorithm classification accuracy of 80% was obtained. C4.5 was applied also in [Hyontai 2012]. The author proposed an oversampling technique in the labeled class with a higher error rate and could achieve a classification accuracy of 78%. Artificial Immune Recognition System (AIRS) with k-NN was applied in [Babu et al. 2015] and obtained classification accuracy of 70%. The authors used AIRS for preprocessing the dataset and then used k-NN as a classifier. Based on the papers that were found for ILPD, the best results was obtained by [Camilleri et al.2014] by using ID3 with classification accuracy of 100% followed by [Liang et al. 2013] that used the genetic algorithm with a classification accuracy of 98%.

In this thesis we additionally applied ensemble techniques and meta-classifiers like bagging, cost-sensitive learning and AdaBoost. The impact of parameters for chosen classifiers and different preprocessing techniques were evaluated. Figure 4.39 shows the best achieved results of the used classifier.

The best classification accuracy of 79.81% was achieved by using bagging classifier. This was done by using decision table for base classifier and10 for bag sizes and number of iterations. The second highest classification accuracy of 73.58% was obtained by using random forest. This was done by using depth sizes of 18, 200 trees and 10% of features.

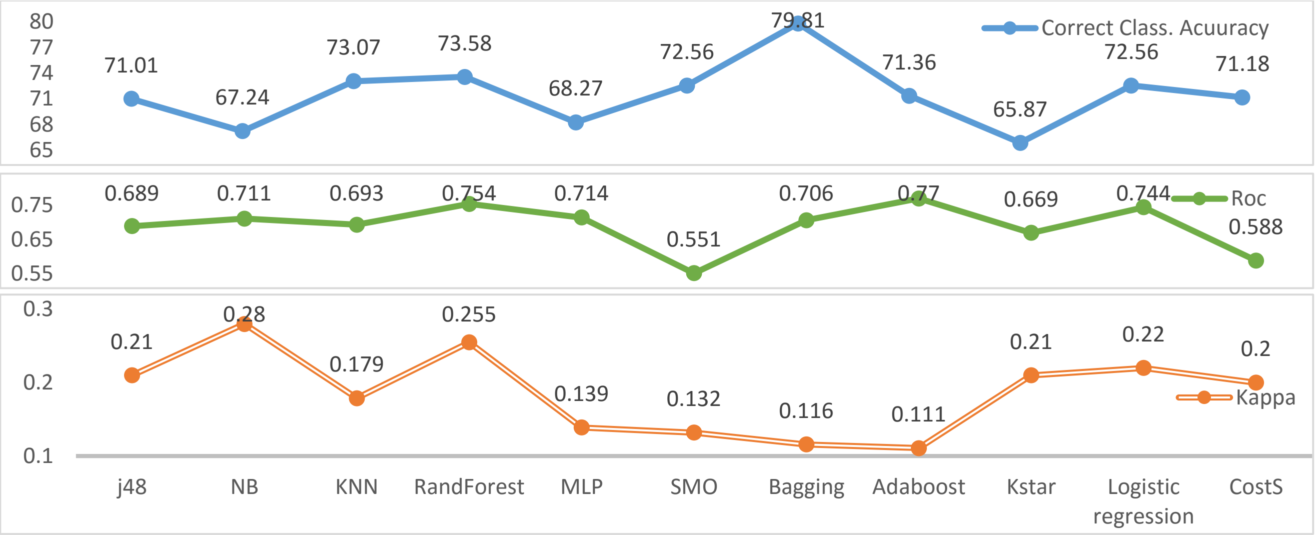


*Figure 4.39- The evaluation performance for ILPD using implementation of various classifiers in Weka*

With respect to the previous works on this dataset, authors tried to improve the performance by applying some feature selection techniques before starting the classification process. They achieved in [Camilleri et al.2014], [Tiwari et al. 2013] and [Liang et al. 2013] higher classification accuracy. While in [Camilleri et al.2014] percentage-split-evaluation method was used, by [Tiwari et al. 2013] the dataset was divided by Gender. We also experimented with different preprocessing techniques and tried to tune classifiers by varying the configuration for significant parameters, which are presented and discussed briefly below.

Regarding the evaluation of k-NN algorithm, the highest classification accuracy was obtained by using:

- Weighting function: 1/distance (see Equation 2.4)
- k = 50

Figure 4.40 shows the performance of k-NN classifier for ILPD using different k values and weighting functions. It indicates that the classification accuracy increased, while the numbers of neighbors raised till 50. In opposite, the kappa statistic value decreased and mean absolute error increased.

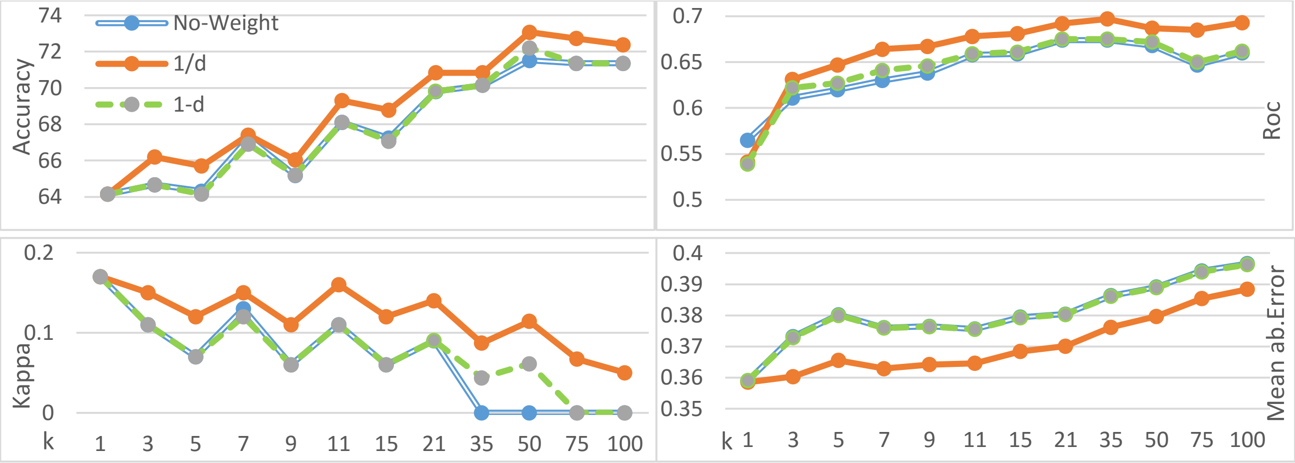


*Figure 4.40- The impact of k and weighting function in k-NN algorithm*

Considering the results using random forest algorithm, the highest classification accuracy was obtained by using:

- 10% to 30% number of features
- 200 trees
- Tree depth size: 18

Figure 4.41 shows the performance of this classifier using different number of features, number of trees and tree depth sizes. The results show that the classifier reached a stable state after using 200 trees or a depth size of 18.

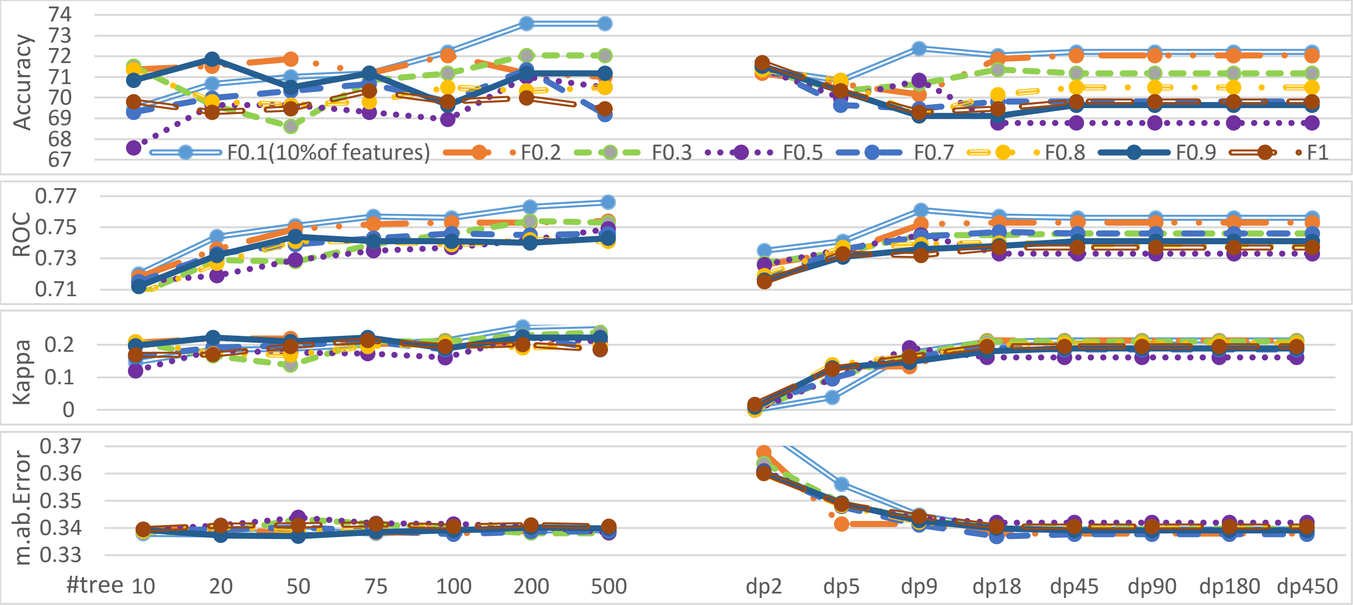


*Figure 4.41 - The impact the number of features against the number of trees and the depth size of a tree in random forest*

Figure 4.42 shows the impact of the number of trees and tree depth sizes on the results of random forest. It indicates that the tree depth size of 18 and higher had the less absolute mean error. The tree depth sizes of 10 performs best for this classifier with more than 20 trees in configurations.

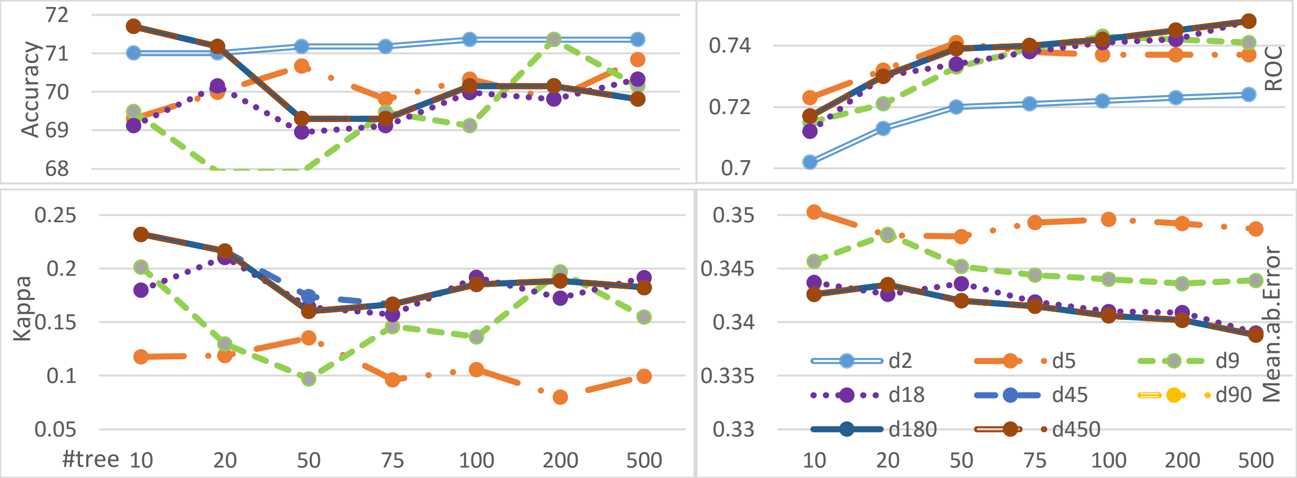


*Figure 4.42- The impact of the depth size of tree against number of trees in the random forest*

Considering the evaluation of bagging, the highest classification accuracy was obtained by using:

- Decision table

- Bag sizes: 10 and 500
- 10 and 5 iterations

Figure 4.43 shows the performance of the classifier with different base classifiers, bag sizes and interation numbers.

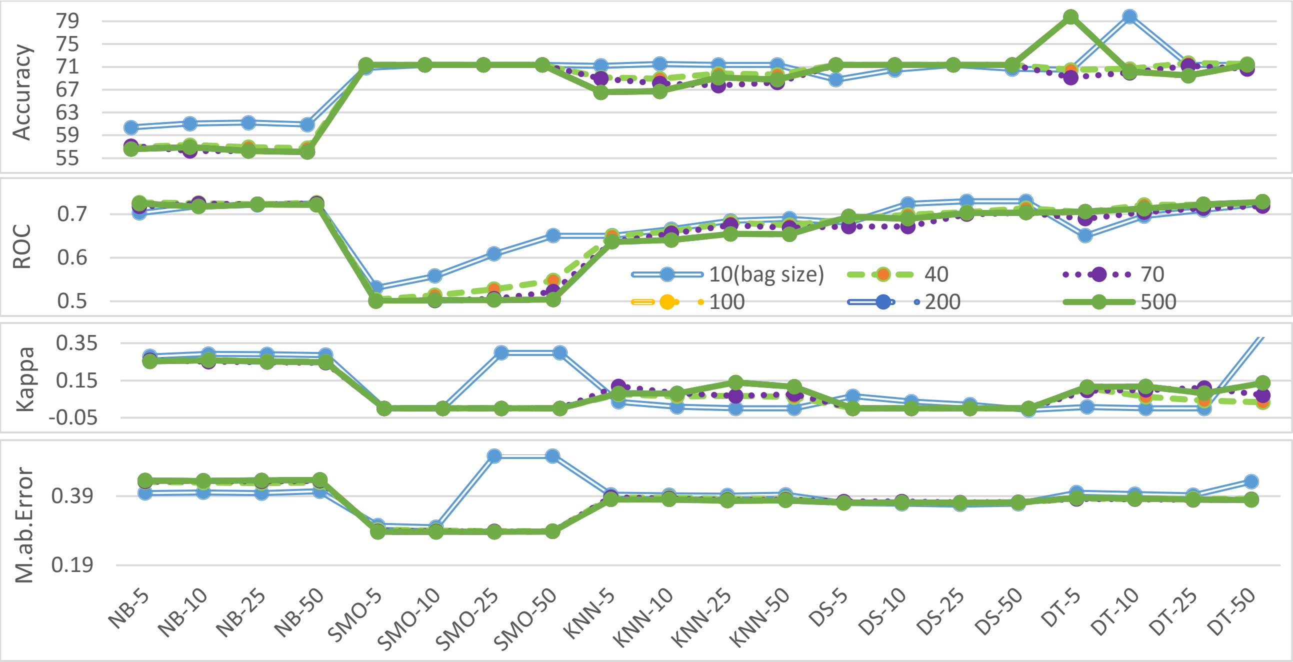


*Figure 4.43- The impact of selected parameters in the bagging algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using AdaBoost, the highest classification accuracy was obtained by using:

- SMO as base classifier
- Weight threshold: 100 and higher
- 25 Iterations and higher

Figure 4.44 shows the performance of the classifier with different configurations for base classifiers, weight thresholds and iteration numbers. The results indicate that the base classifier had the highest impact on the performance. In this experiment the best accuracy was obtained by using SMO base classifier but this algorithm needed more execution time.  The number of iterations had only effect to the decision stump classifier.

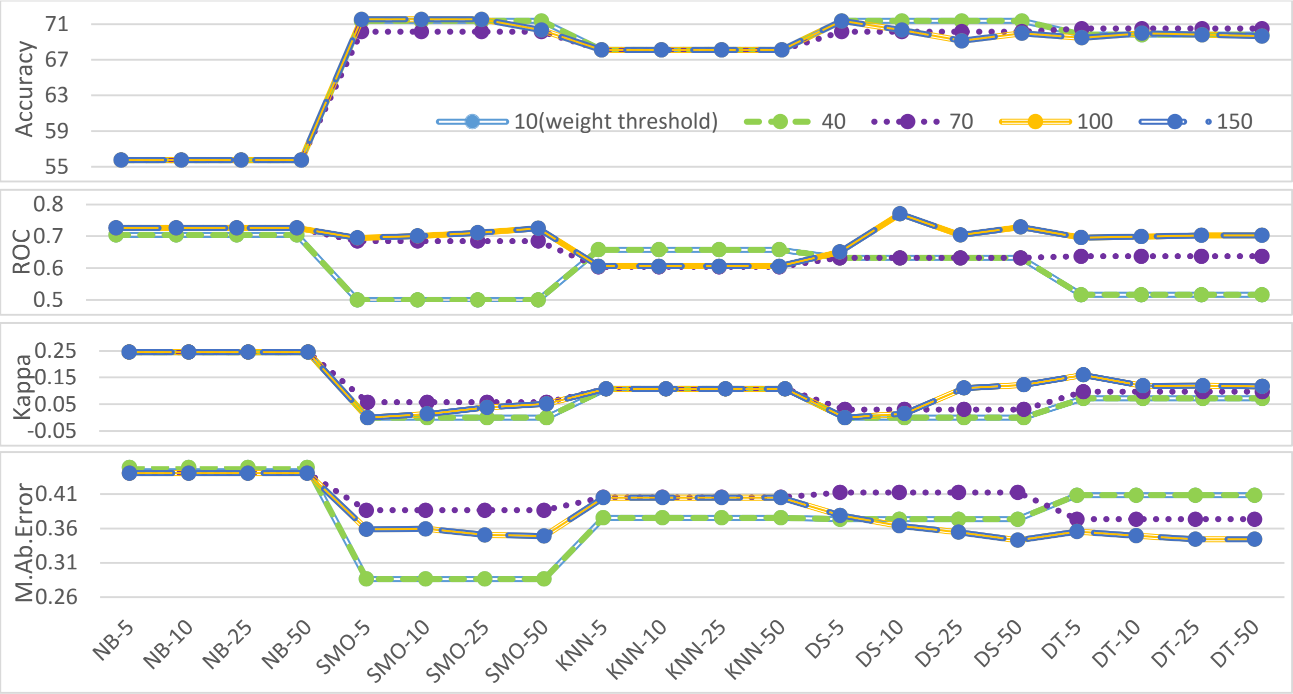


*Figure 4.44- The impact of selected parameters in the AdaBoost algorithm*

NB-5: means that naive bayes classifier and 5 iterations were

Considering the results using J48 classifier, the highest classification accuracy was obtained by using:

- Number of objects (instances) per leaf: 1
- Confidence factor: 0.75

Figure 4.45 shows the performance of J48 classifier with different configurations for the most significant parameters like confidence factor. The results indicate that the number of instances per leaf had the highest impact on the performance followed by the confidence factor.

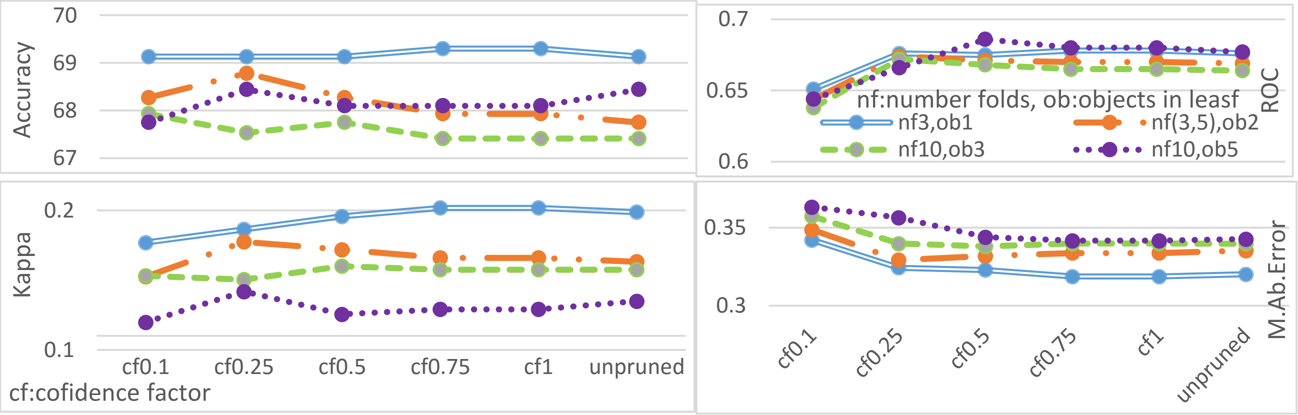


*Figure 4.45- The impact of selected parameters in the J48 algorithm*

Considering the evaluation of SVM Classifier, the highest classification accuracy was obtained by using:

- RBF kernel in SMO and Polynomial kernel in SVM
- Without normalization and standardization of data

Figure 4.46 shows the performance of the SMO and SVM classifier with different kernels. The results indicate that SMO obtained in most of cases better performance in the experiment.

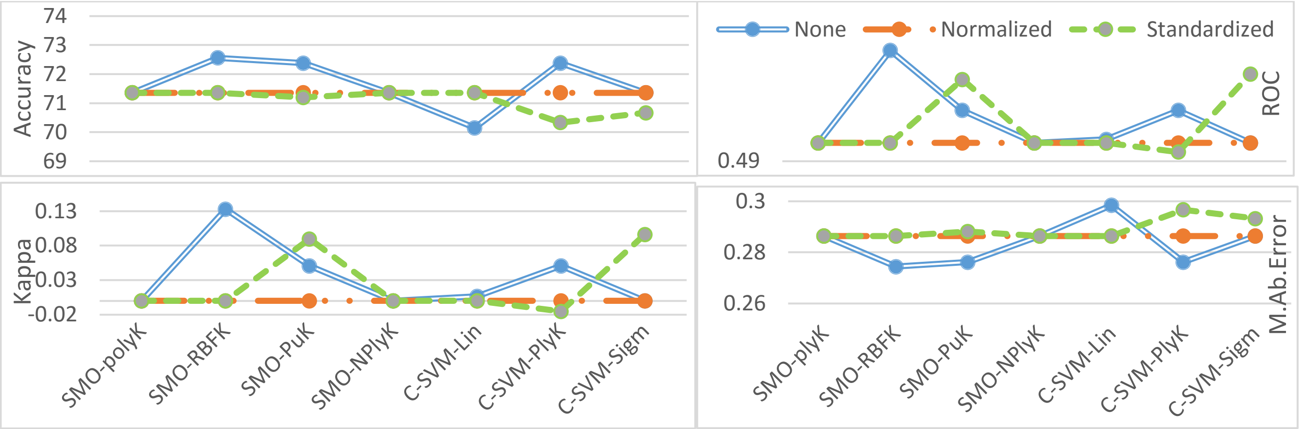


*Figure 4.46- The impact of selected parameters in SVM & SMO algorithms*

Features 6 and 7 of dataset were discretized to four bins using Weka discretization tool. Figure 4.47 shows the performance before and after applying the discretization. The performance of tree base classifiers (random forest and J48) improved in contrast to k-NN and NB.

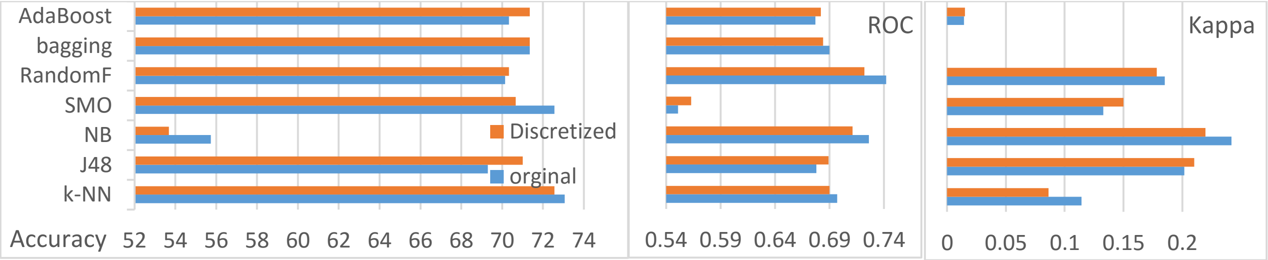


*Figure 4.47- The impact of preprocessing methods in the selected algorithms*

## 4.6 Mammography mass

For Mammographic Mass dataset (MMD) different techniques have been used. Decision tree and case-based reasoning have been applied in [Elter at al.2007]. The ROC was used for evaluation of classification performance and values of 0.89 and 0.87 were obtained respectively. In [Ferrari 2011], a Reliable Support Vector Machine (RSVM) was proposed to improve the classification accuracy. The study combined benefit of SVM algorithm with proposed method in [Museli et al.2007]. The best classification accuracy of 70% was obtained. Bagging, AdaBoost and EM clustering were applied in [Halawani 2012] and authors respectively obtained classification accuracy of 80.7%, 78.4%, and 78.9%. A neural network venn predictor (NN-VP) has been used in [Harris 2011], the author utilized the vann predication framework provided [Vladimir et al.2005] and combined it with neural networks. The performance of the proposed method was compared with standard neural network (NN), resulting in 78.92% classification accuracy in comparison to 78.83 classification accuracy that was obtained with NN. Binary logistic regression (BLR), random forest, C4.5, cost-sensitive classifier, k-NN, logistic regression, MLP and SVM were applied in [Jacob 2012]. The authors evaluated the performance based on classification accuracy and error-rate. The highest accuracy of 91% was obtained by using random tree and c4.5. In [Kathleen et al. 2013] authors applied neutral network training model while the missing values were removed. This method obtained of 89.64% accuracy and the value 0,962 for ROC. Bayesian network was applied in [Kharya et al.2014]. In [Ludwig 2010] the author proposed two genetic programming approaches for classification and applied them on the data set. The recorded performance of both ROC values were 0.859 and 0.860. Decision tree, neural network, and SVM were applied in [M. & E. 2013] with the best classification accuracy of 81.25% using SVM (A ROC value of 0.831). A nearest-prototype-style classifier was used in [Chalmers et al.2014] which was optimized with a genetic algorithm to perform high predictions of positive values (PPV). In [Huang et al. 2012] authors have applied the particle swarm optimizer based on ANN, the adaptive neuro-fuzzy inference system (ANFIS), and case-based reasoning classifier. The highest experimental result of 92.8% classification accuracy was obtained by using ANFIS. In [Elsayad 2010] authors applied two different implementations of bayesian network (tree augmented (TAN) and markov blanket estimation (MBE) learning algorithms) whereby the highest classification accuracy obtained was 87.85% using the Bayesian-MBE. Based on the papers that were found for MMD, the best result was obtained by [Huang et al. 2012] using ANFIS with classification accuracy of 92.8% followed by [Jacob 2012] which used the random tree with a classification accuracy of 91%.

In this thesis we additionally applied different classifiers like k-star and other ensemble techniques. Also the impact of the parameters in selected classifiers and different preprocessing techniques (discretization, feature selection, normalization, standardization and dealing with missing values) were evaluated. Figure 4.48 shows the best performance achieved by selected classifiers.

The best classification accuracy of 88% was achieved by using SMO classifier. This was done by using normalization in the preprocessing phase and applying polynomial kernel for the classifier configuration.

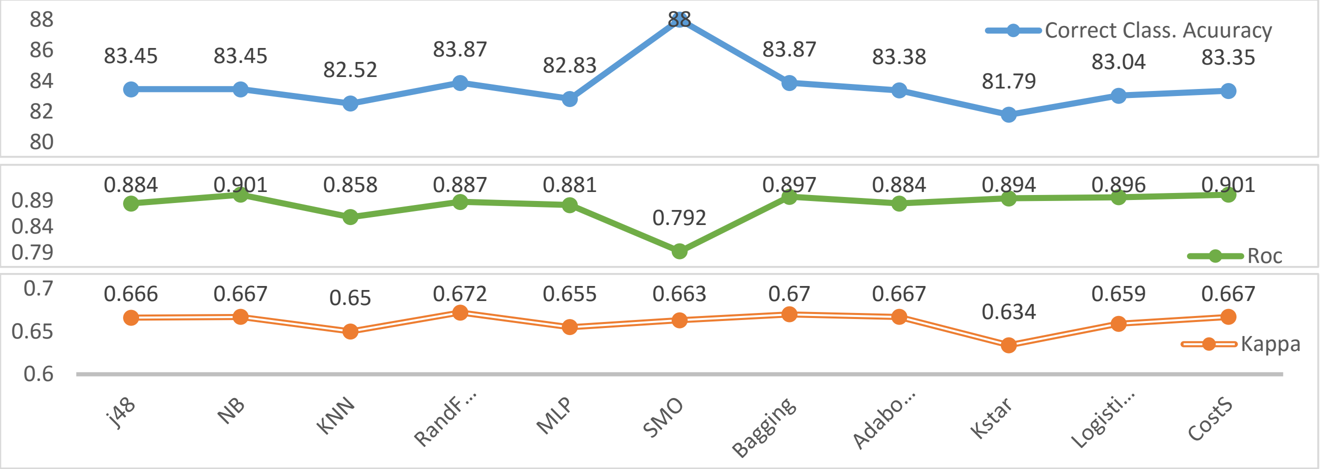


*Figure 4.48- The evaluation performance for MMD using implementation of various classifiers in Weka*

Furthermore, we investigated dealing with missing values (as preprocessing method) which leads to better results. Also we experimented with other preprocessing techniques and tried to tune classifiers by varying the configurations, which are presented and discussed briefly below.

Regarding the results using k-NN classifier, the highest classification accuracy was obtained by using:

- weighting function: 1-distance (see Equation 2.4)
- k = 9

Figure 4.49 shows the performane of k-NN classifier with different configurations of k and weighting functions.

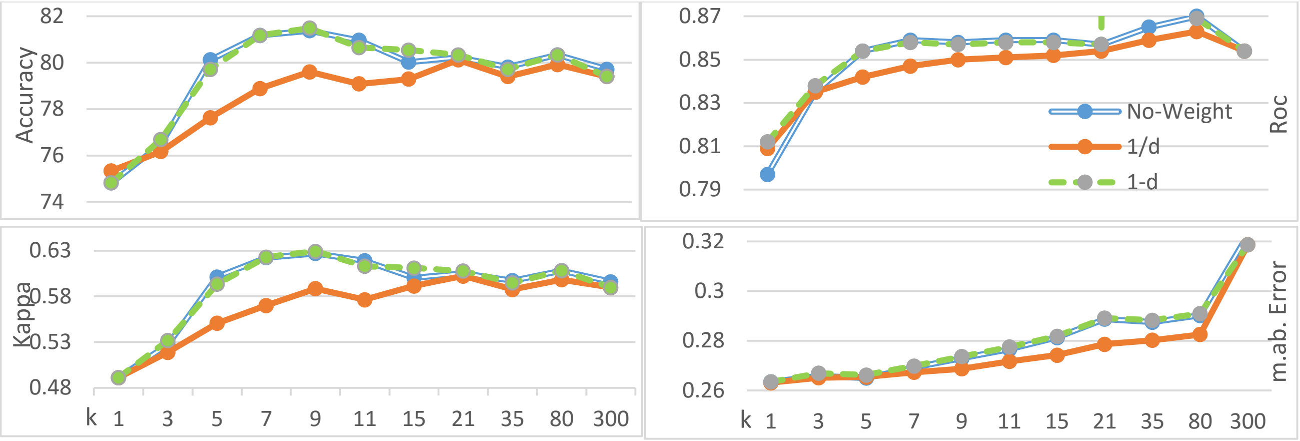


*Figure 4.49 - The impact of k and weighting function in k-NN algorithm*

Regarding the evaluation of random forest classifier, the highest classification accuracy was obtained by using:

- 50% of features
- Number of trees: 50 and 100
- Tree depth sizes: 2 and 5

Figure 4.50 shows the performance of this classifier using a different number of features against different number of trees and tree depth sizes. The results indicate that the number of trees of 50 and higher has a small effect on the performance. It also shows that the tree depth sizes of 2 (followed 5) obtained the highest accuracy.

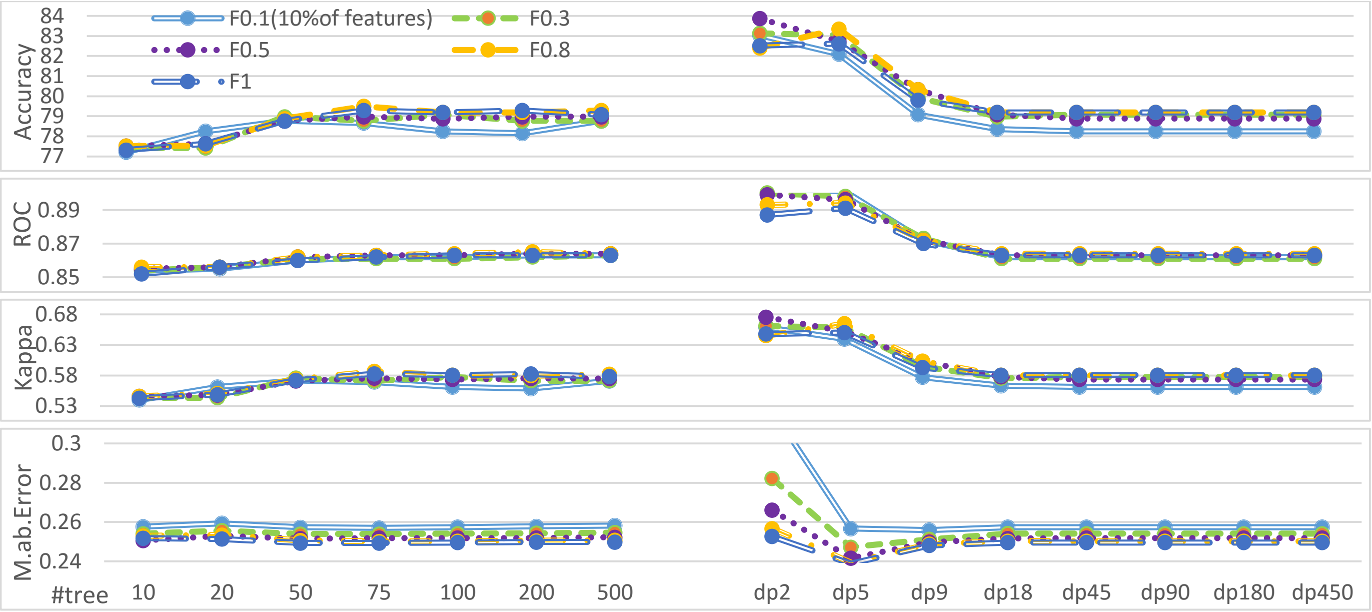


*Figure 4.50 - The impact the number of features against the number of trees and the depth size of a tree in the random forest*

Figure 4.51 shows the impact of the number of trees and tree depth sizes of the results of random forest. It is obvious that a lower depth size (2 and 5) was a better configuration in this dataset.

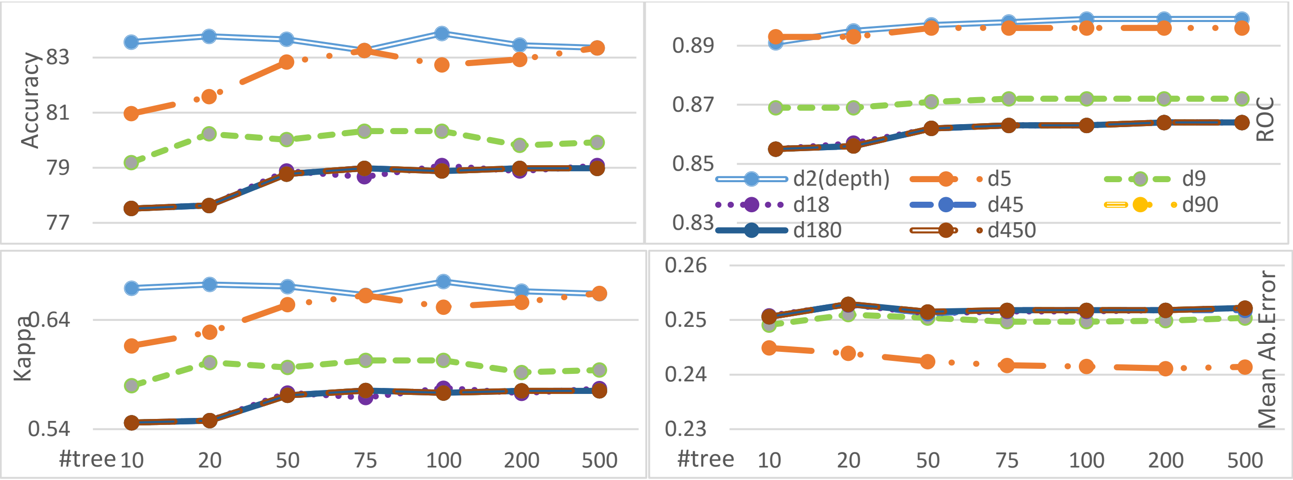


*Figure 4.51- The impact of the depth size of tree against number of trees in the random forest*

Considering the results using bagging, the highest classification accuracy was obtained by using:

- Decision table classifier
- Bag sizes of 500
- 25 iterations

Figure 4.52 shows the performance of this classifier by using different configurations including base classifiers, bag sizes and iteration numbers. The results indicate that the base classifier had the highest imapct on the performance. It also shows that a higher number of iterations gave better performance.

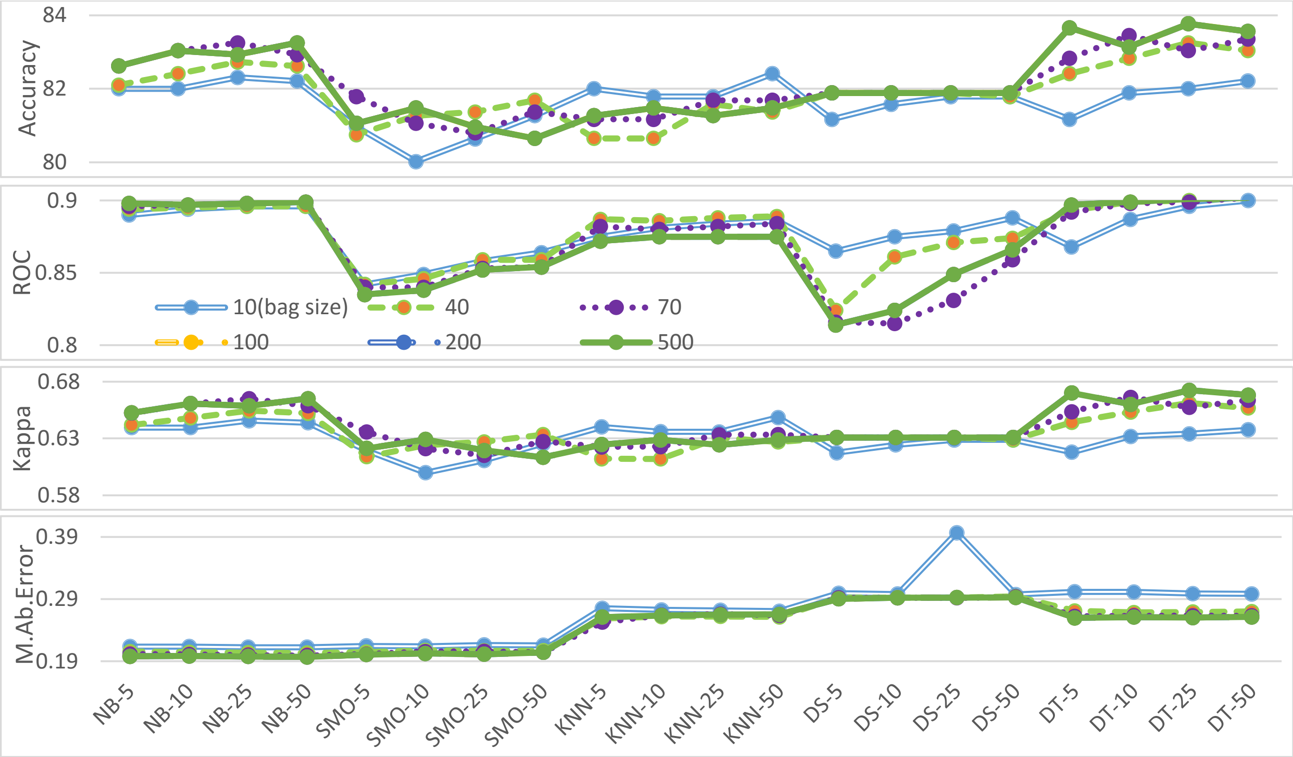


*Figure 4.52- The impact of selected parameters in the bagging algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the evaluation of AdaBoost, the highest classification accuracy was obtained by using:

- Naive bays classifier
- Weight threshold: 100 and higher
- 5 Iterations

As the results are shown in Figure 4.53, the base classifiers had the highest impact on the performance of this classifier. The results show interestingly different impact on performance for SMO and k-NN with varying configurations.

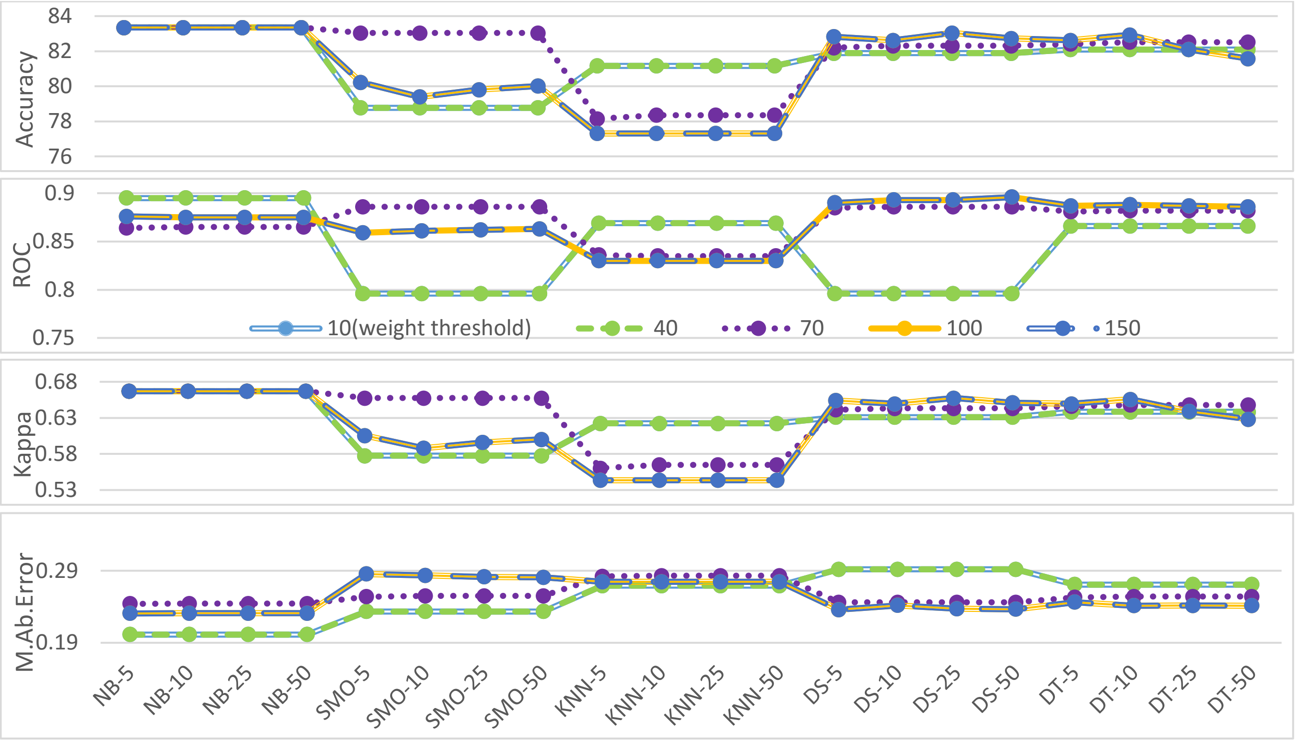


*Figure 4.53- The impact of selected parameters in AdaBoost algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using J48 classifier, the highest classification accuracy was obtained by using:

- Number of objects (instances) per leaf: 5
- Confidence factor of 0.5 and unpruned

Figure 4.54 shows the performance of this classifier using different configurations for the parameters. The results indicate that the confidence factor had the highest impact on the performance followed by number of instances per leaf.

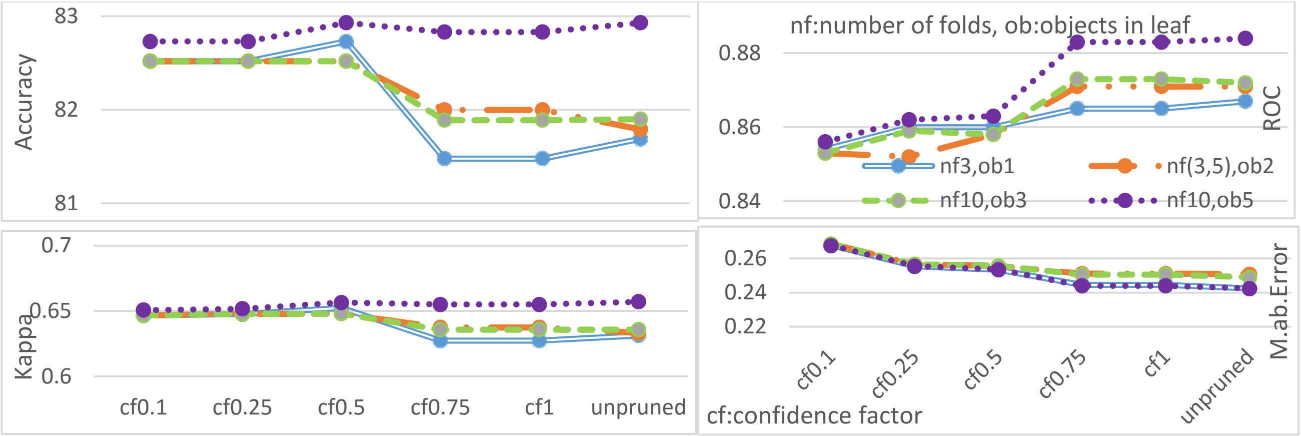


*Figure 4.54 - The impact of selected parameters in the J48 algorithm*

Considering the evaluation of SVM, the highest classification accuracy was obtained by using:

- Polynomial kernel in SMO
- normalization or standardization of data

Figure 4.55 shows the performance of SMO and SVM classifiers with different kernels.

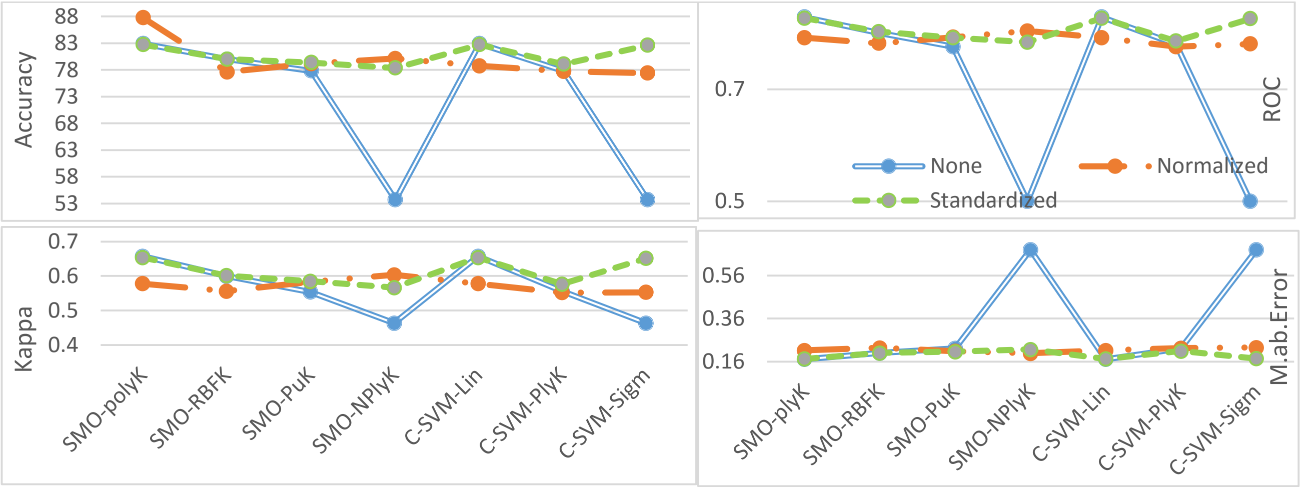


*Figure 4.55- The impact of selected parameters in SVM & SMO algorithms*

Figure 4.56 shows the results of this classification using the various techniques applied for dealing with missing values. The replacement the missing value with mean value of attributes lead to a weaker results in comparison to other methods.

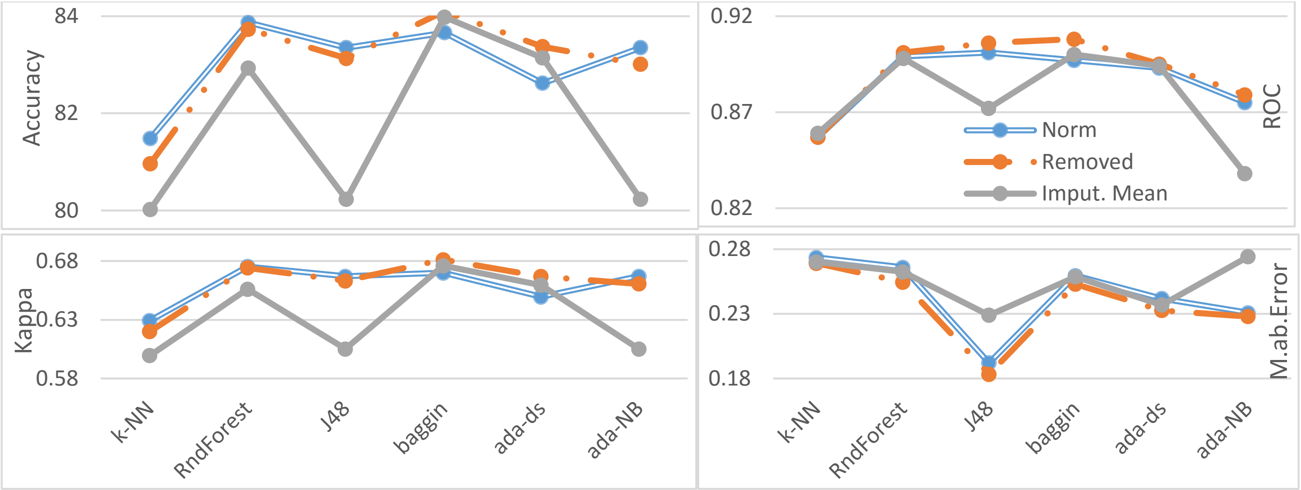


*Figure 4.56- The impact of dealing with missing values in the selected algorithms*

Figure 4.57 shows the performance before and after applying the preprocessing. We discretized the second feature of this dataset using twelve bins in the Weka discretization tool. It improved the accuracy of the tested classifiers up to 1%. The result also shows that the discretization enhanced the results in classifiers like k-NN, NB and J48.

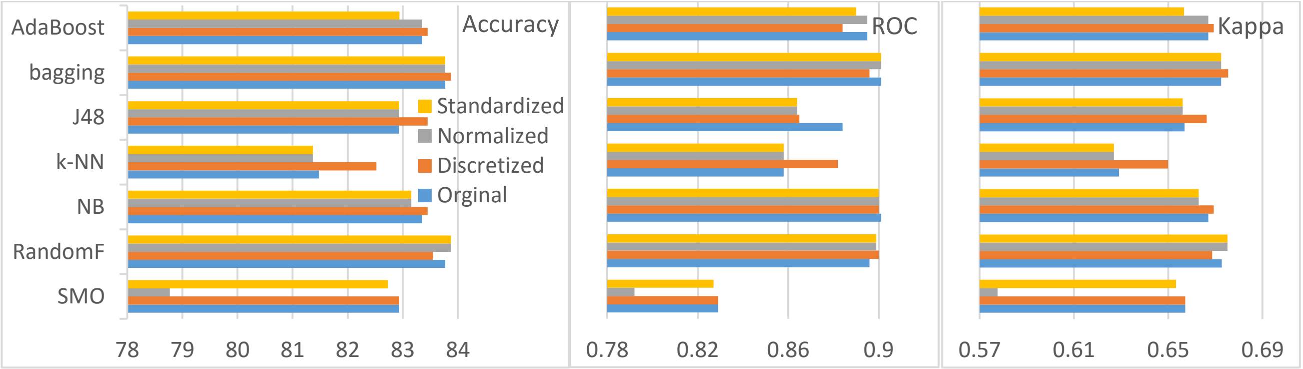


*Figure 4.57- The impact of preprocessing methods in the selected algorithms*

## 4.7 Spect Heart

For SPECT heart dataset different techniques have been used. CLIP3 (Cover learning using integer programming [Cios et al.1997]) was applied in [Kurgan et al. 2001] with 84% classification accuracy. The four bayesian classifiers (Markov blank bayesian classifier (MBBN), normal bayesian network (BN), naive bayes and tree-augmented naive bayes(TANB)) have been applied in [Maddaen 2002]. The highest classification accuracy of 80.75% was obtained by using MBBN. An ensemble learning technique based on active example select (EAES) was applied in [Oh et al. 2011] with the help of random undersampling and random over-sampling techniques. The highest classification accuracy of 79.8% was obtained by using EAES. The bayesian network was applied in [Onisko et al. 2013]. The decision tree with and without help of laplace correction was applied in [Eyke et al. 2008]. In [Huang et al. 2005] authors applied a new proposed dynamic ensemble re-construction using bagging and AdaBoost techniques. The new rank boosting algorithm "RandBoost" was utilized to improve ranking performance on chosen classifier in ensemble learner. The proposed method achieved the highest ROC value of 0.903 using AdaBoost. Decision stump, decision tree, random forest, and naive bayes were applied in [Elazmeh et al. 2006] and achieved the highest classification accuracy of 76.5%. The SVM was applied in [Peng et al. 2010], which used sequential forward floating search (SFFS) and proposed features selection method. The highest ROC value of 0.832 was obtained by the proposed method. SVM was applied in [Polat et al. 2009] using the proposed kernel F-score feature selection method and obtained accuracy rate of 83.46%. For the proposed method in [Lianq et al. 2009] authors applied bayesian networks using stochastic approximation Monte Carlo. In [McSherry 2011] the author applied a new approach of conversational case-based reasoning called iNN that improved classification accuracy. This approach achieved classification accuracy of 82.3%.

Based on the papers that were found for this dataset, the best result was obtained by [Kurgan et al. 2001] using CLIP3 (84% classification accuracy) followed by [McSherry 2011] that achieved technique a classification accuracy of 83.46%.

In this thesis we additionally applied the k-star and other ensemble techniques. Also the impact of parameter configurations for chosen classifiers and different preprocessing techniques were evaluated. Figure 4.58 shows the best performance achieved for selected classifiers. The best classification accuracy was achieved by using AdaBoost. This was done using decision stump classifier and 25 iterations.

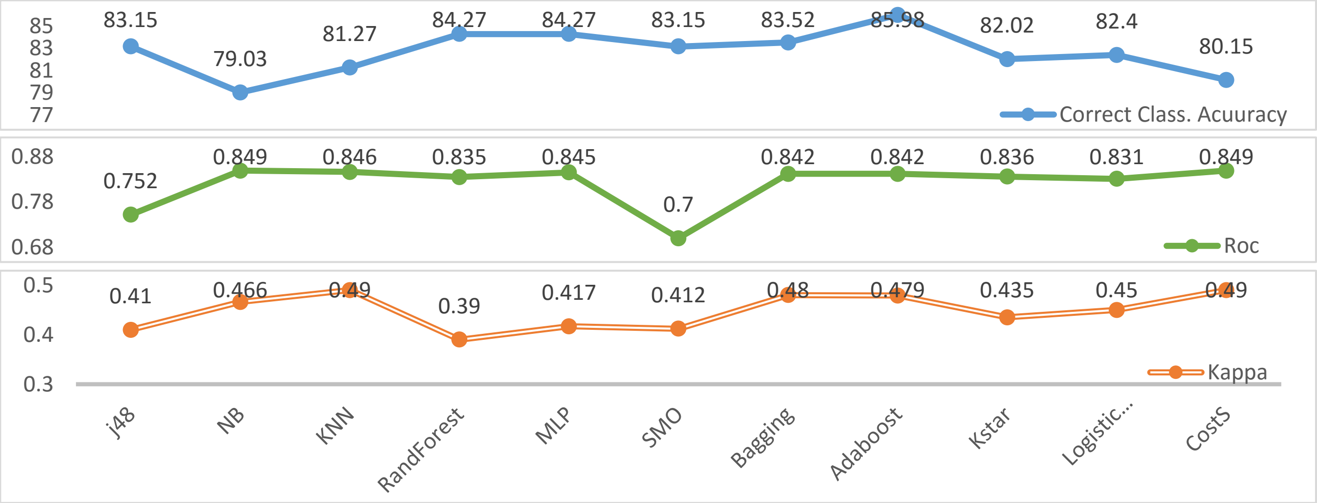


*Figure 4.58- The performance for SPECT Dataset using implementation of various classifiers in Weka*

With respect to the previous works on this dataset, authors tried to improve the performance by applying some feature selection techniques before starting the classification process. We achieved higher classification accuracy of 85.98 by using AdaBoost algorithm. Also we experimented with different preprocessing techniques and tried to tune classifiers by varying the configuration for significant parameters, which are presented and discussed briefly below:

Regarding the evaluation of k-NN classifier, the highest classification accuracy was obtained by using:

- Weighting function: 1 - distance (see Equation 2.4)
- k = 7

Figure 4.59 shows the results of k-NN classifier using different configurations for k values and weighting functions. According to the k parameter, the k = 120 obtained the highest classification accuracy, but k = 7 was the best configuration regarding all measurement methods (accuracy, kappa statistic, ROC and mean absolute error).

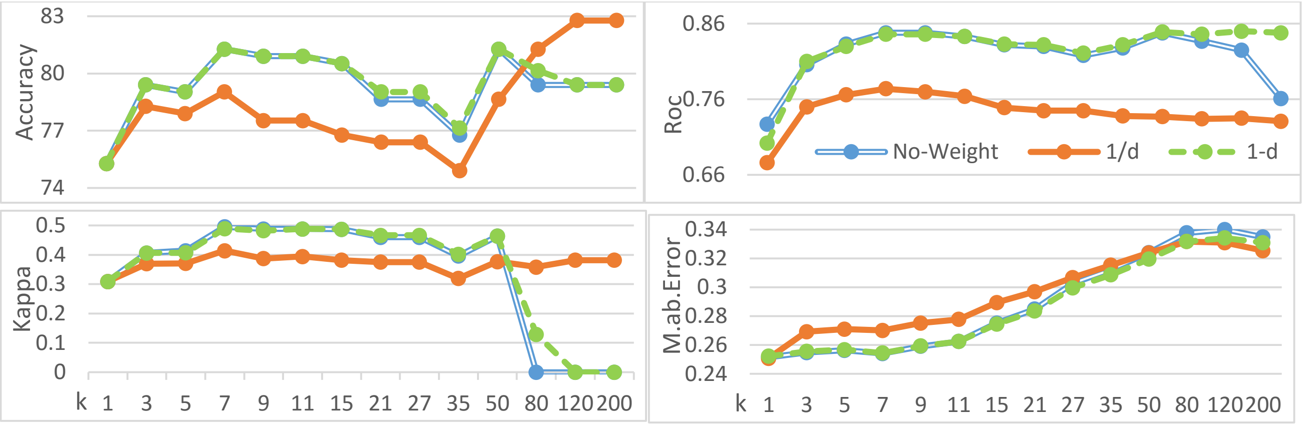


*Figure 4.59 - The impact of k and weighting function in the k-NN algorithm*

Regarding the results using random forest classifier, the highest classification accuracy was obtained by using:

- 10% of features
- 20 trees
- Tree depth sizes: 5

Figure 4.60 shows the performance of this classifier using a different number of features against different number of trees and tree depth size. The results indicate that the maximum depth of tree is 18.

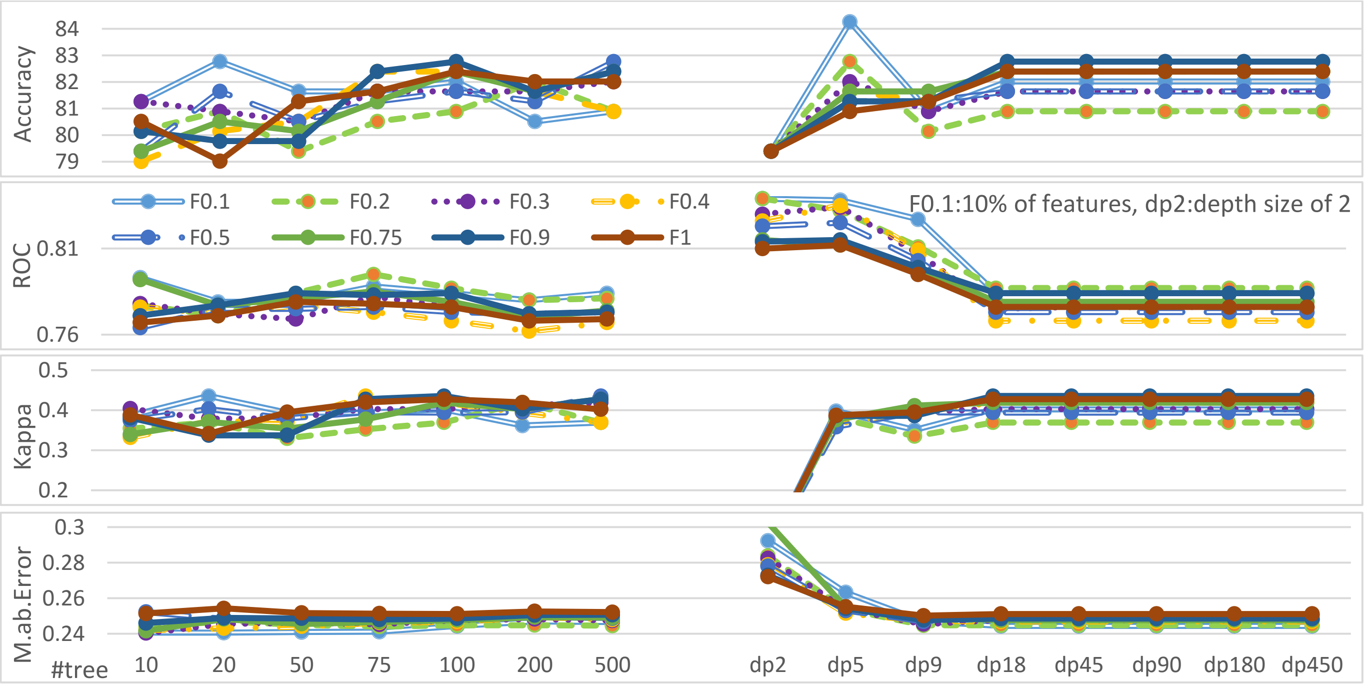


*Figure 4.60 - The impact the number of features against the number of trees and the depth size of a tree in random forest*

Figure 4.61 shows the impact of the number of trees and tree depth sizes of the results of random forest. It indicates that the lower depth size of 5 gave better classification accuracy and depth of 5 had the best results for ROC.

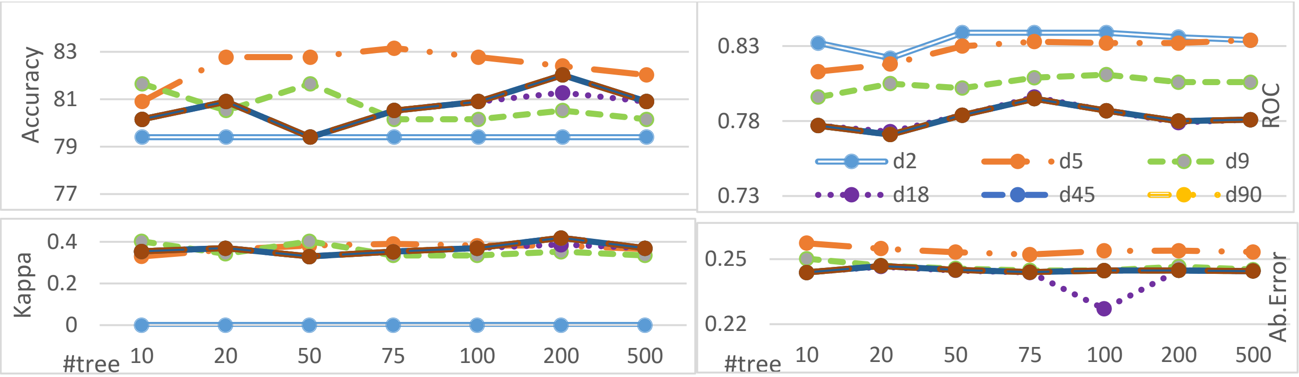


*Figure 4.61- The impact of the depth size of tree against number of trees in the random forest*

Considering the evaluation of bagging, the highest classification accuracy was obtained by using:

- SMO classifier
- Bag size: 10 and 70
- 50 iterations

Figure 4.62 shows the performance of bagging with different configurations including base classifiers, bag sizes and iteration numbers. The results indicate that a higher number of iterations improved the performance in three of four tested classifiers. It also shows that the bag size of 10 obtained low performance in NB in opposite to SMO.

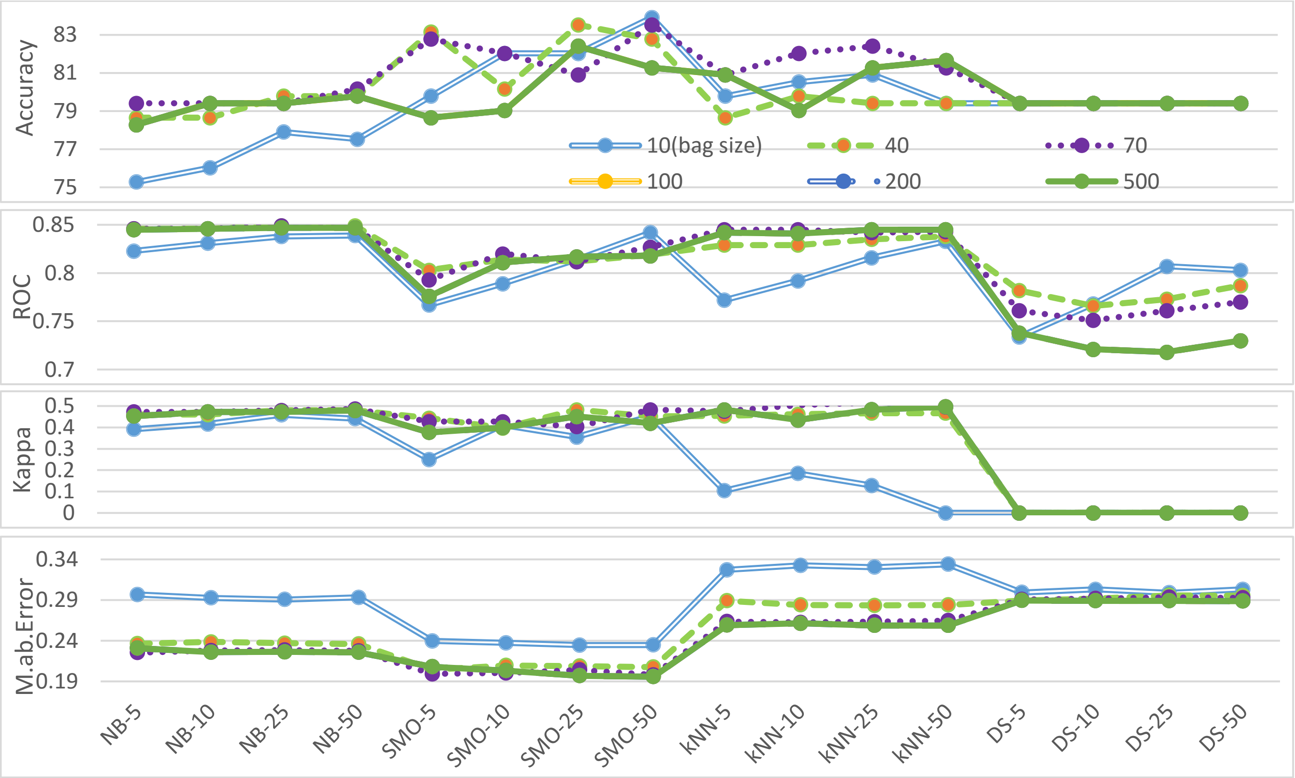


*Figure 4.62- The impact of selected parameters in the bagging algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using AdaBoost, the highest classification accuracy was obtained by using:

- Decision stump classifier

- Weight thresholds: 100
- 25 Iterations

Figure 4.63 shows the performance of classifiers on AdaBoost with different configurations. The results indicate that weight threshold and the number of iterations had the highest impact on the performance.

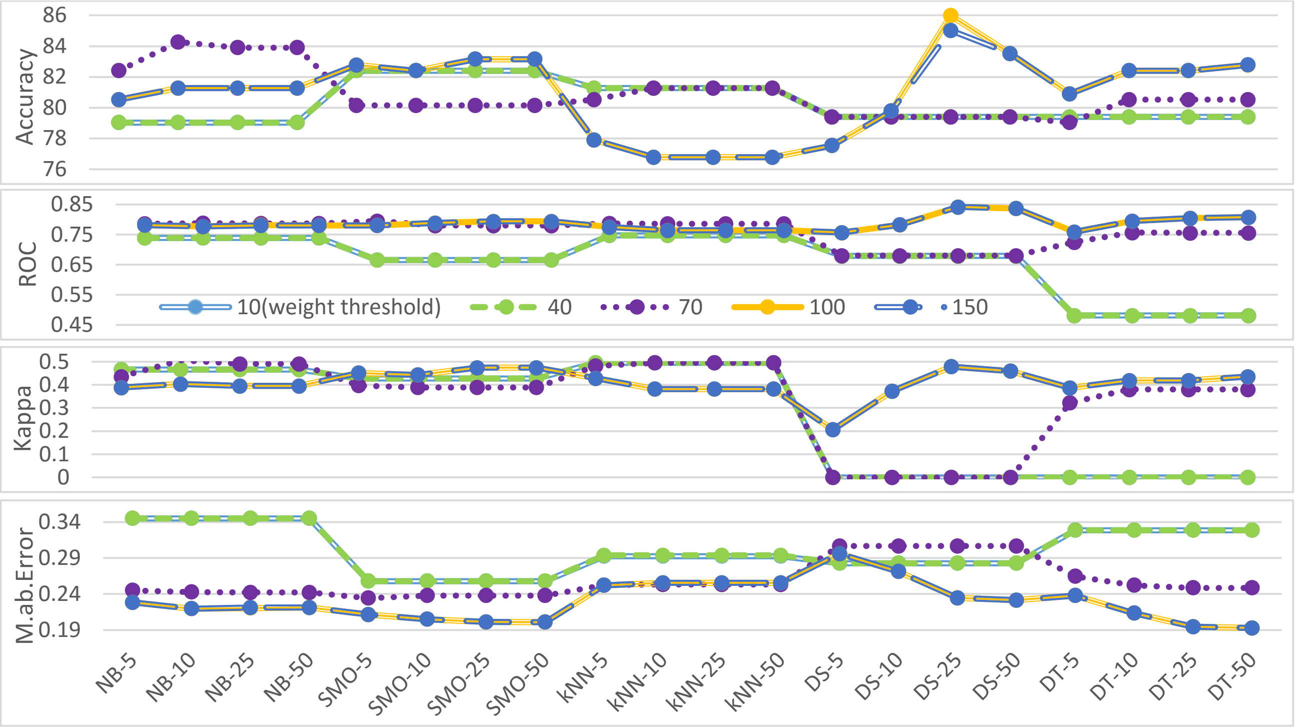


*Figure 4.63 - The impact of selected parameters in AdaBoost algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using J48 classifier, the highest classification accuracy was obtained by using:

- Number of objects (instances) per leaf: 5
- Unpruned and confidence factor of 0. 5 and higher

Figure 4.64 shows the performance of the J48 classifier with different configurations. The results indicate that the confidence factor (effect of pruning) had the highest impact on the performance followed by the number of instances per leaf.

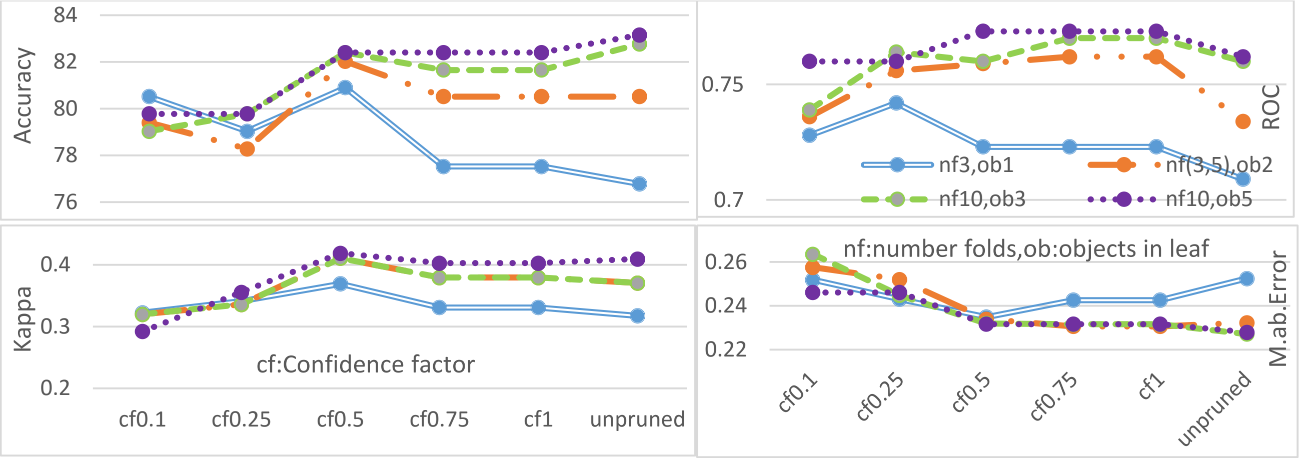


*Figure 4.64- The impact of selected parameters in the J48 algorithm*

Considering the evaluation of SVM, the highest classification accuracy was obtained by using:

- Polynomial kernel in SMO
- Normalization of data

Figure 4.65 shows the performance of the SMO and SVM classifiers with different kernels. In the experiment, the SMO classifier obtained better performance in most cases.

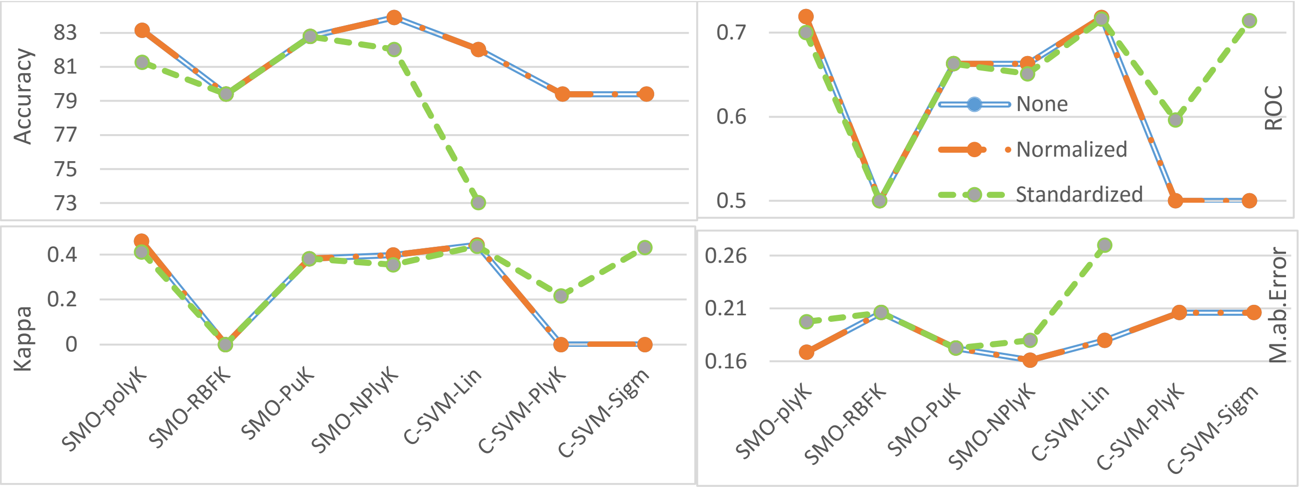


*Figure 4.65- The impact of selected parameters in SVM & SMO algorithms*

## 4.8 Thoracic surgery

For Thoracic surgery (TSD) different techniques have been used by researchers. Ensemble SVM (cost-sensitive) was applied in [Maciej et al. 2013] and resulted in classification accuracy of 60%. In [Maciej 2014] the author applied Cart, MLP, C4.5, bagging, random forest, SVM, Cost-sensitive SVM, AdaBoost, logistic regression and obtained the best classification accuracy by using SVM of 82%. A boosted SVM was applied in [Maciej et al. 2014]. The AdaBoost, deep belief network, random forest, SVM, perceptron (learning) network, k-NN, naive bayes and linear discriminant were used in [Drot at al. 2014], the best classification accuracy of 85.19% was obtained by using SVM. In [Sindhu et al. 2014] authors applied decision stump, random forest, J48, naive Bayes and OneR. In [McBride et al. 2014] authors have applied proposed a cost-sensitive classification called clearance under threshold classification (CUT). The MLP, J48 and naive bayes were used in [Danjuma 2015]. The best classification accuracy of 82.3% was obtained using MLP. The decision trees, naive bayes and SVM were applied in [Nachev et al. 2015]. The best classification accuracy of 79.4% was obtained with SVM. The decision tree and neural network were used in [Jinyan et al. 2015] with the best obtained classification accuracy of 80.2%. Based on the papers that were found for this dataset, the best results were obtained by [Drot at al. 2014], which used the random forest with a classification accuracy of 88%.

In this thesis we additionally applied k-star and additional ensemble techniques. Figure 4.66 shows the best performance achieved by selected classifiers.

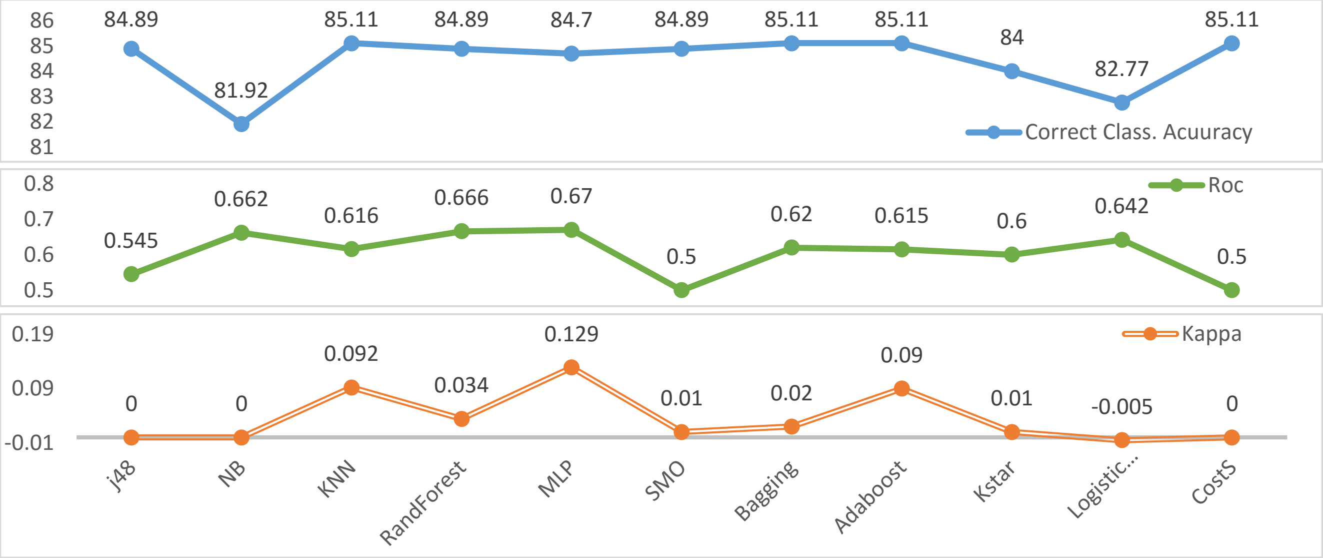


*Figure 4.66- The performance for TSD using implementation of various classifiers in Weka*

We experimented with different preprocessing techniques and tried to tune classifiers by varying the configurations, which are presented and discussed briefly below:

Regarding the evaluation of using k-NN classifier which is shown in Figure 4.67. The highest classification accuracy was obtained by using:

- Weighting function: 1 – distance (see Equation 2.4)
- k = 5

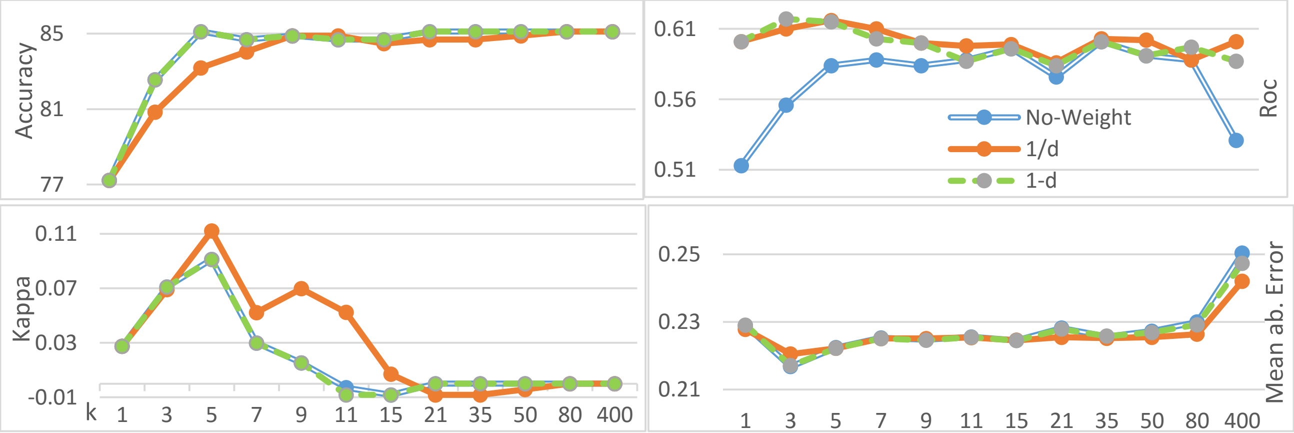


*Figure 4.67 - The impact of k and weighting function in k-NN algorithm*

Regarding the results using the random forest classifier, the highest classification accuracy was obtained by using:

- Number of the features: 10%-30%
- Number of trees: 50 and higher
- Tree depth sizes: between 2 and 9

Figure 4.68 shows the performance of this classifier using different number of features against different number of trees and the tree depth sizes. It is obvious from the results that by increasing the number of features the classification accuracy was reduced. And also lower tree depth sizes resulted in higher classification accuracy.

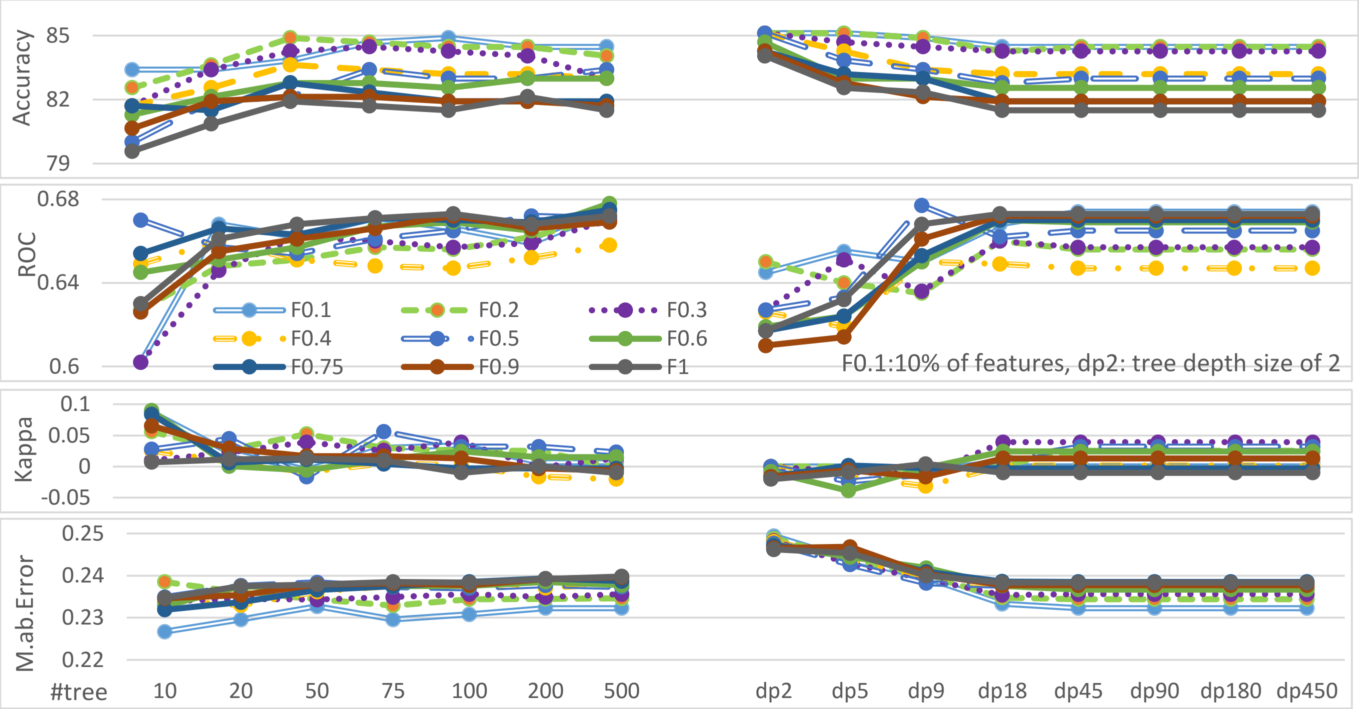


*Figure 4.68 - The impact of the number of features against the number of trees and the depth size of a tree in random forest*

Figure 4.69 shows the impact of the number of trees and tree depth sizes of the results. A lower tree depth size achieved a higher classification accuracy.

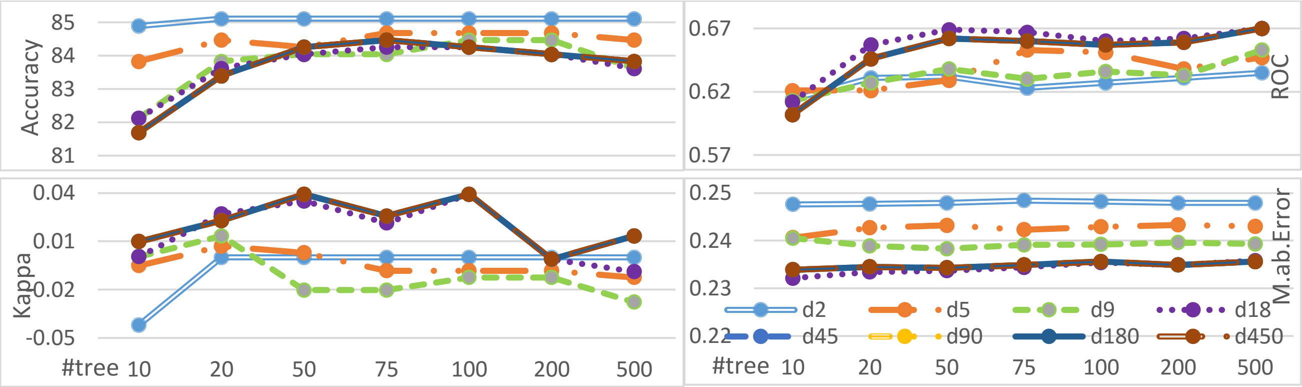


*Figure 4.69- The impact of the depth size of tree against number of trees in the random forest algorithm*

Considering the evaluation of bagging, the highest classification accuracy was obtained by using:

- k-NN and decision table classifier

- Bag size of 10 and higher
- 5 iterations

Figure 4.70 shows the impact of different base classifiers, bag sizes and iteration numbers in the bagging. The results indicate that a higher number of iterations improved the classification performance for naive bayes and for all base classifiers regarding the ROC values.

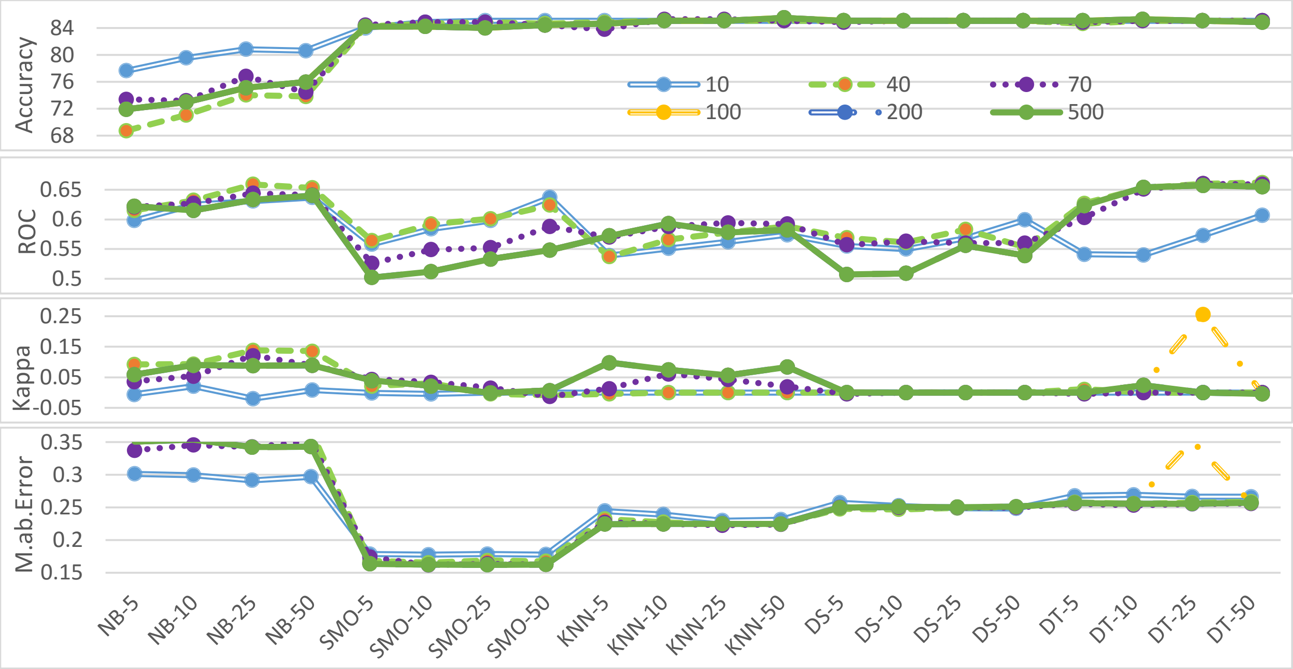


*Figure 4.70- The impact of selected parameters in bagging algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the results using AdaBoost, the highest classification accuracy was obtained by using:

- k-NN classifier
- Weight threshold: 40
- 5 Iterations

Figure 4.71 shows the performance of the AdaBoost by using different base classifiers, weight thresholds and iteration numbers. The results indicate that the base classifier had the highest impact on the performance followed by the weight threshold.

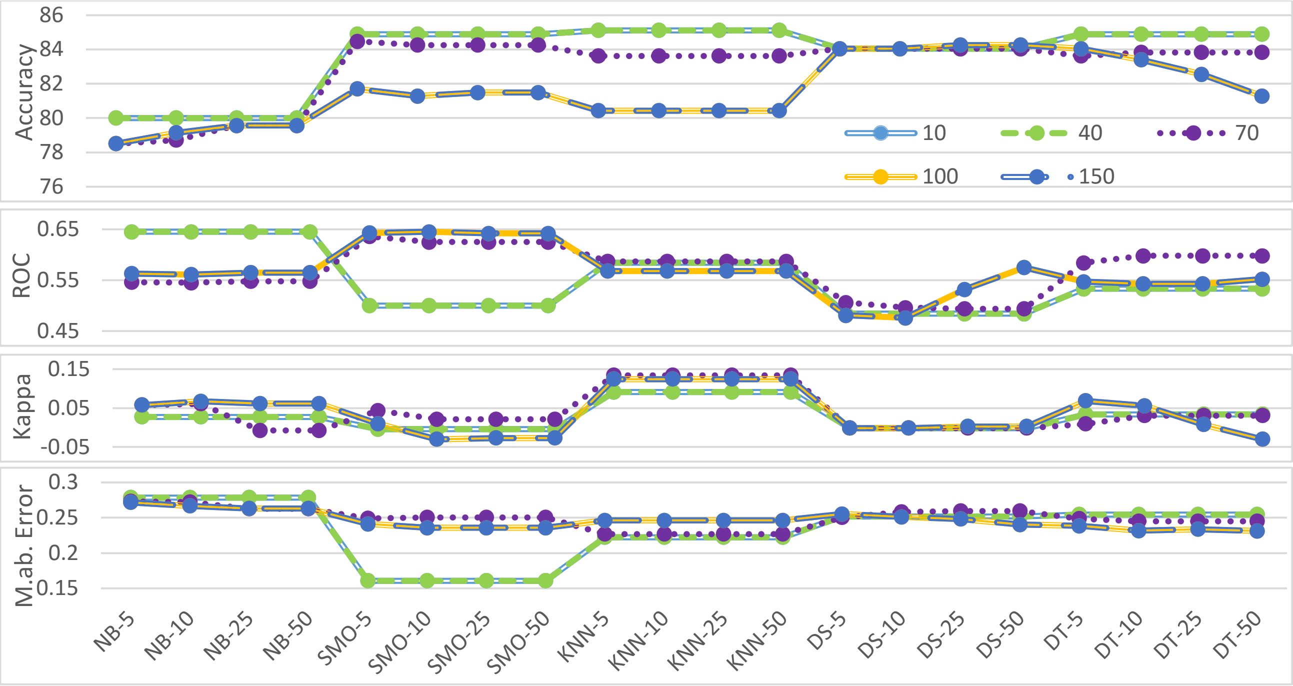


*Figure 4.71- The impact of selected parameters in AdaBoost algorithm*

NB-5: means that naive bayes classifier and 5 iterations were used

Considering the evaluation of J48, the highest classification accuracy was obtained by using:

- Number of objects (instances) per leaf: 1 and higher
- Confidence factor: 0.1

Figure 4.72 shows the performance of the J48 classifier with different configurations for number of objects in leafs, number of folds and confidence factors. The results indicate that the pruning parameter (confidence factor) had the highest impact on the performance followed by the number of instances per leaf.

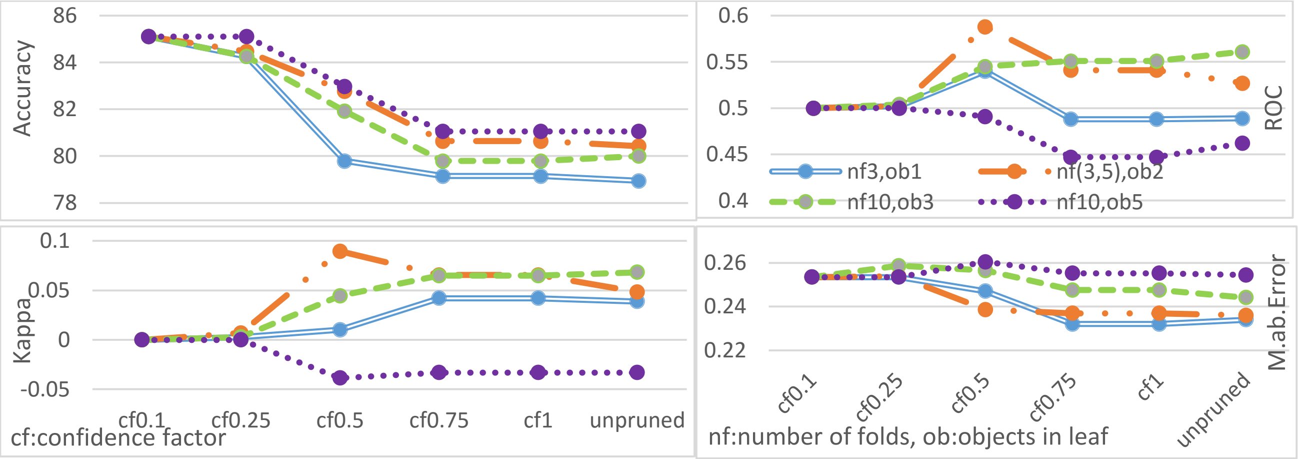


*Figure 4.72- The impact of selected parameters in the J48 algorithm*

Considering the evaluation of SVM, the highest classification accuracy was obtained by using:

- Puk (Pearson) kernel in SMO and polynomial kernel in SVM
- Normalization

Figure 4.73 shows the performance of the SMO and SVM classifiers with different kernels. In the experiment, the results indicate that SMO obtained usually better performance.

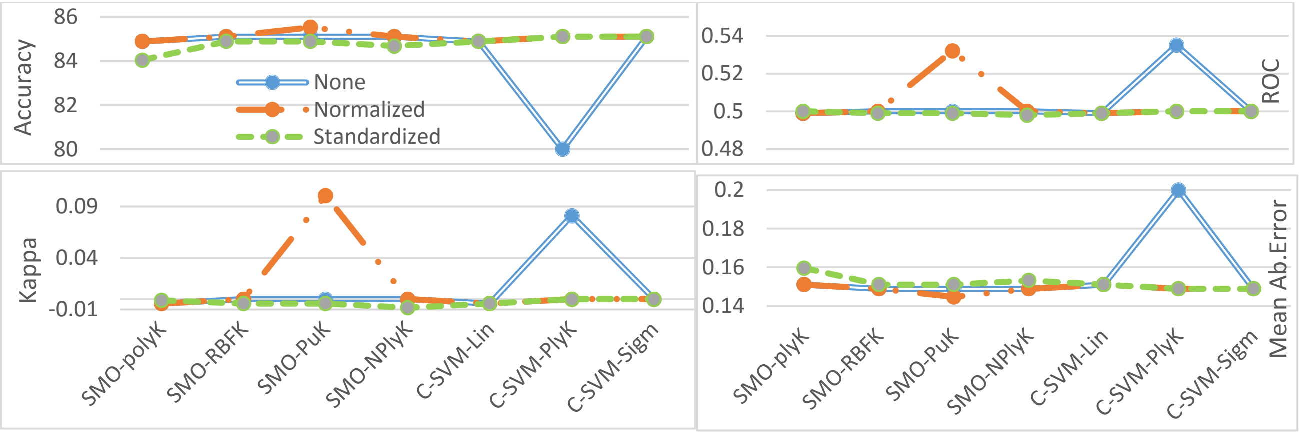


*Figure 4.73- The impact of selected parameters in SVM & SMO algorithms*

Figure 4.74 shows the performance of classifiers before and after applying the preprocessing. We discretized the second feature of this dataset using twelve bins in the Weka discretization tool. It improved the accuracy of NB classifier up to 4 %.

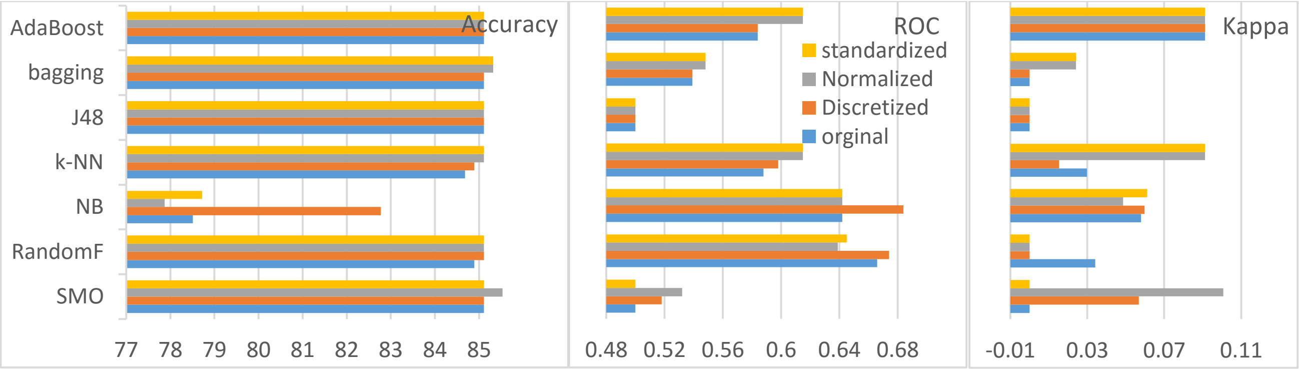


*Figure 4.74- The impact of preprocessing methods in the selected algorithms*

# Chapter 5

# Summary of results

In this chapter the results obtained from the experiments in Chapter 4 are compared and summarized. The comparison of techniques is based on three performance measures: classification accuracy, ROC and kappa statistic value.

One of the aims in this thesis was to find the proper classifier for medical datasets. For achieving this aim, the result of each individual experiment was analyzed and compared. Figure 5.1 shows the results of all used classifiers on all data sets. Results show that no technique outperforms all other techniques in all data sets. However, ensemble learning techniques like AdaBoost and random forest give in several data sets best results.

The naive bayes classifier usually achieved the lowest classification accuracy. However, regarding the ROC recorded results this classifier achieved good values in four of data sets.

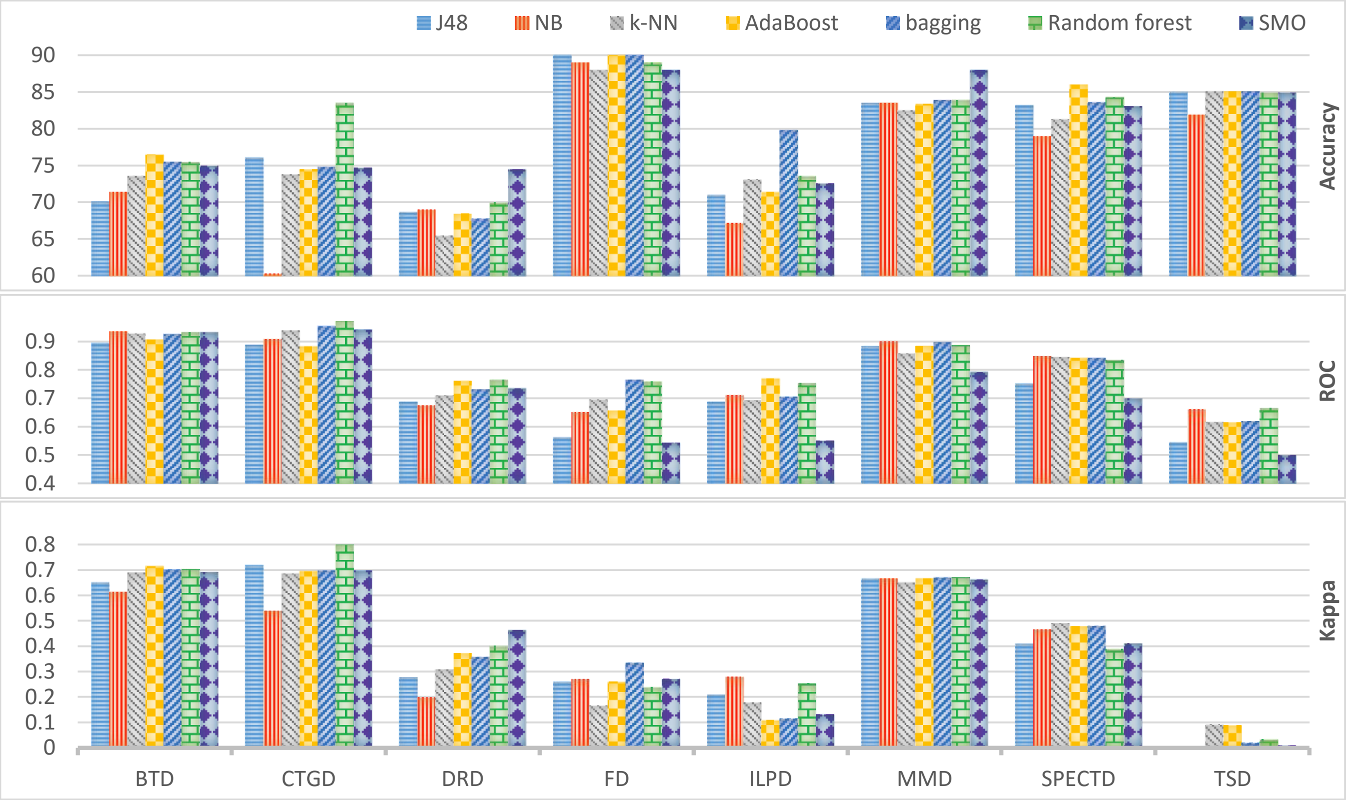


*Figure 5.1- The summary of performance of each individual classifier in each dataset*

In our experiments we also analyzed the significant parameters for the selected classifiers. With respect to all investigation in the experiments, based on the variation of various configuration parameters for each classifier, following parameters for each classifier turned out and proved, by in depth comparison, to be the most significant ones, regarding their impact on the classifier's performance.

In k-NN classifier, the most significant parameter was the k value (number of neighbors) and the second significant one was the weighting function. In decision tree (J48) classifier, the most significant one was the confidence factor (pruning factor) and the next one was number of objects per leaf. In random forest, the number of trees, number of features and tree depth size were the most significant parameters. In bagging, the base classifier, number of iterations and bag size were the most significant parameters. In AdaBoost, the base classifier, number of iterations and weight threshold were the most significant parameters. In SVM and SMO, the kernel function was one of the most significant parameters.

Figure 5.2 shows the maximum difference in the performance of a classifier, based on different parameter configurations on selected data sets. The results indicate that the impact of parameters was not important in each data set. For the first two data sets the tuning of parameters improved significantly the results, whereas for some data sets (e.g., fertility) the results improved only slightly. The highest impact on the classifier performance was in the SMO classifier followed by the k-NN classifier.

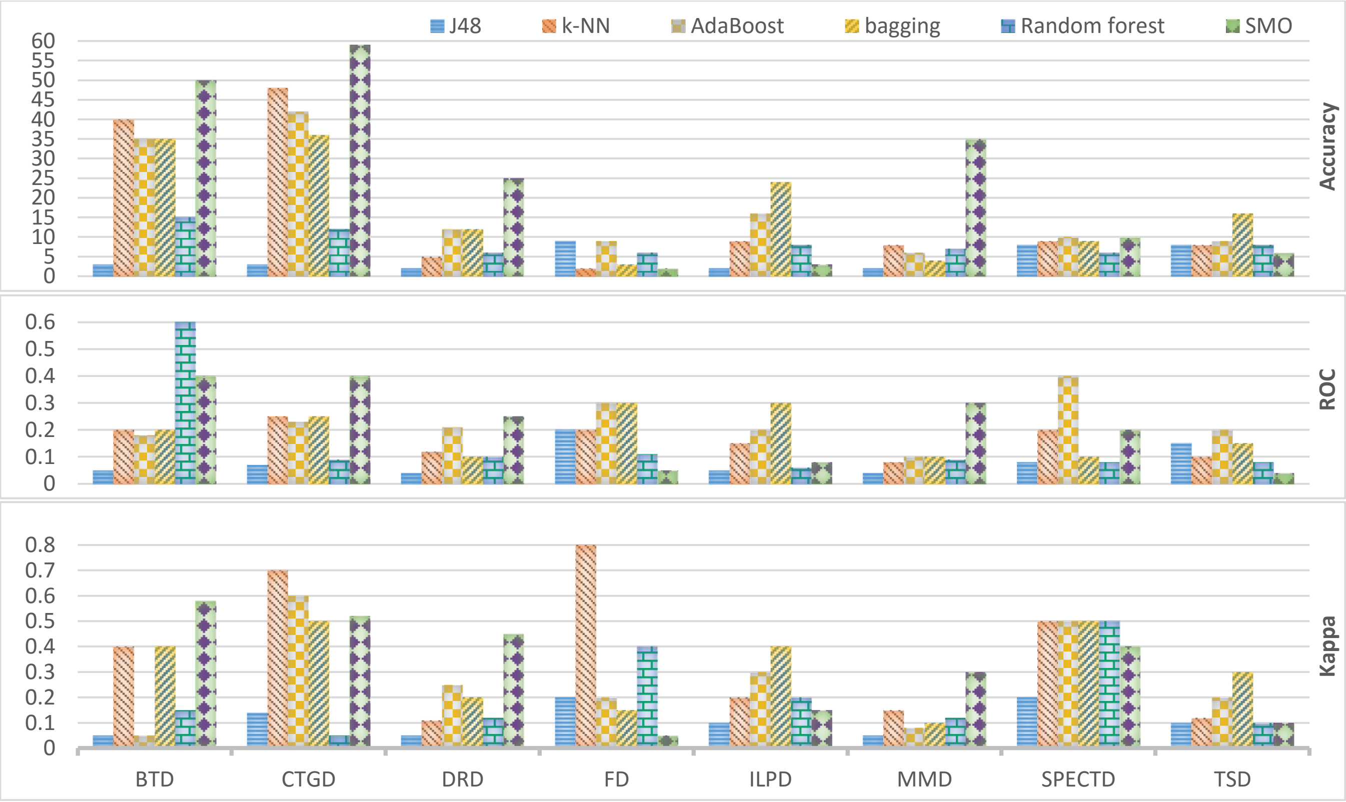


*Figure 5.2- The maximum differences in the classifier performance based on various parameter configurations*

Figure 5.3 shows the maximum differences in the performance of each classifier after applying the preprocessing techniques. The results indicate that the impact on the performance is depended on the characteristics of each data set. The used preprocessing techniques did not affect the results for the SPECT heart data set, because this data set contains only attributes with binary values. The results indicate that the preprocessing techniques had more impact on the SMO classifier. The lowest impact of preprocessing techniques was observed for the k-NN and J48 classifiers.

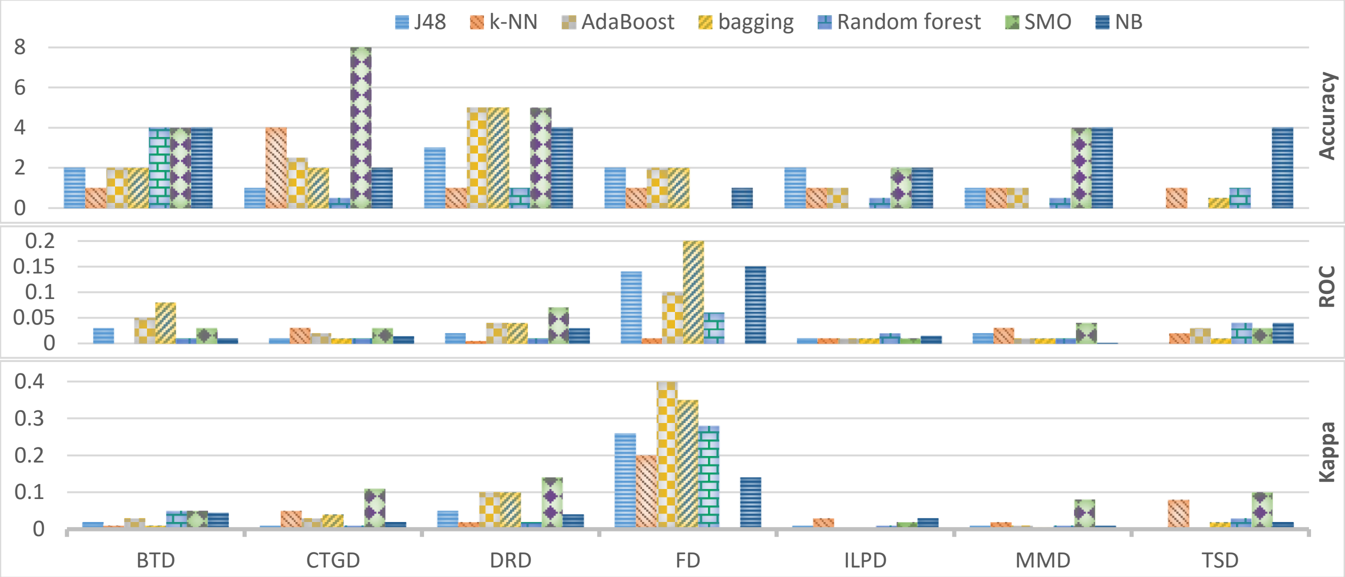


*Figure 5.3- The maximum impact on the performance of classifiers based on the selected preprocessing method*

In this thesis, different missing values imputation techniques such as standard mean imputation and removing missing instance were studied and evaluated. The mean imputation improved the performance for the bagging and AdaBoost classifiers. Although this technique improved the results, the use of other more advanced techniques [Schafer et al. 2002] could still improve the results.

# Chapter 6

# Conclusions and future work

The focus of this thesis was on the application of machine learning techniques in medical data sets. We investigated various classification techniques and evaluated the impact of classifier configuration and preprocessing techniques.

We analyzed and selected different medical data sets from the UCI repository. The focus was particularly on data sets for which is not easy to achieve high classification accuracy. The following data sets were selected: breast tissue, cardiotocography, diabetic retinopathy, fertility, indian liver, mammography mass, spect heart and thoracic surgery.

We reviewed the machine learning techniques that have been used by other researchers for these data sets. We then experimentally evaluated on the same data sets the following classifiers: AdaBoost, bagging, decision tree (J48), k-NN, naive bayes, random forest and SVM. Furthermore, we investigated parameters in each classifier and performed experiments with various configurations. The impact of the preprocessing techniques on selected datasets was investigated in details.

The experimental results showed that no classification technique was able to obtain very good results for all data sets. However, ensemble learning techniques like AdaBoost and random forest obtained best results in several data sets. The use of preprocessing techniques and parameter tuning was very important to achieve good performance for the most machine learning algorithms.

Overall, we can conclude that to obtain good results for such data sets the experiments with different classification techniques, different parameter configurations and different preprocessing techniques are needed. Results can be improved significantly by applying different techniques.

For the future work, it would be interesting to investigate other machine learning techniques such as deep learning. Furthermore, the investigation of feature selection techniques for these data sets would help to identify the most important features.

# Bibliography

[Aha et al. 1991] Aha David W., Dennis Kibler and Marc K. Albert: Instance-based learning algorithms. *Machine learning* 6.1:37-66 (1991).

[Antal et al. 2014] Antal B. and Andras H. An ensemble-based system for automatic screening of diabetic retinopathy. *Knowledge-Based Systems* 60: 20-27 (2014).

[Babu et al. 2014] Babu Prasad, Swapna k., Balakrishna T. and Venkateswarulu N.B. An implementation of hierarchical clustering on Indian Liver Patient Dataset*. International Journal of Emerging Technologies in Computational and Applied Sciences* 8.6: 543-547 (2014).

[Babu et al. 2015] Babu, M. S., and Somesh Katta: Artificial immune recognition systems in medical diagnosis*. 6th International Conference of Software Engineering and Service Science (ICSESS)* pp. 1082-108, (2015).

[Bahramirad et al. 2013] Bahramirad Shay, Aouache Mustapha, and Maryam Eshraghi: Classification of liver disease diagnosis: A comparative study. *Second International Conference on Informatics and Applications (ICIA),* pp. 42-46 (2013).

[Bates et al.3003] Bates D. W., Kuperman G. J., Wang S., Gandhi T., Kittler A., Volk L.,and Middleton B. Ten commandments for effective clinical decision support: making the practice of evidence-based medicine a reality. *Journal of the American Medical Informatics Association* 10.6: 523-530 (2003).

[Bellazzi et al.2008] Bellazzi Riccardo and Blaz Zupan: Predictive data mining in clinical medicine: current issues and guidelines. *International journal of medical informatics* 77.2: 81-97 (2008).

[Berner 2007] Berner Eta S.: Clinical decision support systems. *New York: Springer Science+ Business Media, LLC* (2007).

[Bose et al.2001] Bose Indranil and Radha K. Mahapatra: Business data mining - a machine learning perspective. *Information & Management* 39.3: 211-225 (2001).

[Breiman 1996] Breiman Leo. Bagging predictors. *Machine learning 24.2*: 123-140 (1996).

[Breiman 2001] Breiman Leo. Random forests. *Machine learning 45.1*: 5-32 (2001).

[Camilleri et al.2014] Camilleri, M., F. Neri, and M. Papoutsidakis: An Algorithmic Approach to Parameter Selection in Machine Learning using Meta-Optimization Techniques. *WSEAS Transactions on Systems* 13, 202-213 (2014).

[Chalmers et al.2014] Eric Chalmers, Marcin Mizianty, Eric C. Parent, Yan Yuan, Edmond Lou: Toward maximum-predictive-value classification. *Pattern Recognition* 47.12: 3949-3958 (2014).

[Cios et al.1997] Cios Krzysztof J., Daniel K. Wedding and Ning Liu. CLIP3: cover learning using integer programming. *Kybernetes* 26.5: 513-536 (1997).

[Cios et al.2002] Cios Krzysztof J., and G. William Moore. Uniqueness of medical data mining. *Artificial intelligence in medicine* 26.1:1-24 (2002).

[D., 2015] Daliri, M. R. Combining extreme learning machines using support vector machines for breast tissue classification. *Computer methods in biomechanics and biomedical engineering* 18.2: 185-191, (2015).

[Danjuma 2015] Danjuma Kwetishe Joro, Performance Evaluation of Machine Learning Algorithms in Post-operative Life Expectancy in the Lung Cancer Patients. *arXiv*:1504.04646 (2015).

[Diogo et al. 2000] Ayres-de-Campos Diogo, Bernardes J., Garrido A., Marques-de-Sa J. and Pereira-Leite L. SisPorto 2.0: a program for automated analysis of cardiotocograms. *Journal of Maternal-Fetal Medicine* 9.5: 311-318 (2000).

[Drot et al. 2014] Peter DROT, and S. M. Zdenek. Comparative study of machine learning techniques for supervised classification of biomedical data. *Acta Electrotechnica et Informatica* 14.3: 5-10 (2014).

[Dudani 1976] Dudani Sahibsingh A., The distance-weighted k-nearest-neighbor rule*, IEEE Transactions on Systems, Man, and Cybernetics* 4: 325-327 (1976)

[Dufourq et al. 2013] Dufourq, E., & Pillay, N. : A Comparison of Genetic Programming Representations for Binary Data Classification. *In Third World Congress on Information and Communication Technologies (WICT)* on Dec 15 pp. 134-140 *IEEE*, (2013).

[E. & S., 2007] Matthias Elter, Prof. Dr. Rüdiger Schulz-Wendtland: UCI machine learning repository, https://archive.ics.uci.edu/ml/datasets/Mammographic+Mass , (2007).

[Elazmeh et al. 2006] Elazmeh William, Nathalie Japkowicz and Stan Matwin. Confidence Interval for the difference in classification error*. American Association for Artificial Intelligence*. WS06-06-008 (2006).

[Elsayad 2010] Elsayad Alaa M. :Predicting the severity of breast masses using Bayesian networks. *The 7th International Conference on Informatics and Systems (INFOS 2010),* pp.1-9 *IEEE* (2010).

[Elter at al.2007] Elter M., R. Schulz-Wendtland, and T. Wittenberg: The prediction of breast cancer biopsy outcomes using two CAD approaches that both emphasize an intelligible decision process. *Medical physics* 34.11: 4164-4172 (2007).

[Ertugrul et al.2014] Ertuğrul Ömer Faruk, and Yılmaz Kaya: A detailed analysis on extreme learning machine and novel approaches based on ELM. *American Journal of Computer Science and Engineering* 1.5: 43-50 (2014).

[Eyke et al. 2008] Hüllermeier Eyke and Stijn Vanderlooy, An empirical and formal analysis of decision trees for ranking. *Philipps Universität Marburg, Marburg, Germany*, *Tech. Rep. Comput. Sci. Series* 56 (2008).

[Fawcett 2006] Fawcett Tom. An introduction to ROC analysis. *Pattern recognition letters* 27.8: 861-874 (2006).

[Fayyad et al.1993] Fayyad U. and Irani Keki B. Multi-interval discretization of continuous-valued attributes for classification learning*. Proceedings of IJCAI*, pp. 1022-1027 (1993).

[Ferrari 2011] Ferrari Enrico, and Marco Muselli : Implementing reliable learning through Reliable Support Vector Machines. *Foundations of Computational Intelligence* (FOCI 2011):100-106 *IEEE*, (2011).

[Freund et al.1997] Yoav Freund and Robert E. Schapire: A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* 55.1:119–139, (1997).

[Gardner et al.1998] Gardner Matt W., and S. R. Dorling: Artificial neural networks (the multilayer perceptron) a review of applications in the atmospheric sciences. *Atmospheric environment* 32.14: 2627-2636 (1998).

[Geisser 1993] Geisser Seymour. Predictive inference. Vol. 55. *CRC press* (1993).

[Gil et al.] David Gil, and Jose Luis Girela : UCI machine learning repository, https://archive.ics.uci.edu/ml/datasets/Fertility , (2013).

[Gil Mendez et al.2012] David Gil Méndez, Jose Luis Girela, Joaquin De Juan, M. Jose Gomez-Torres, Magnus Johnsson: *Predicting seminal quality with artificial intelligence methods. Expert Syst. Appl.* 39.16: 12564-12573 (2012).

[H. & M. 2014] Hashem Esraa M., and Mai S. Mabrouk: A Study of Support Vector Machine Algorithm for Liver Disease Diagnosis. *American Journal of Intelligent Systems* 4.1:9-14 (2014).

[H., 2012] Sug Hyontai: Data Mining in Medicine Domain Using Decision Trees-The Case of CART and C4. 5. *Dermatology* 366.35: 6 (2012).

[Halawani 2012] Halawani S. M., M. Alhaddad, and A. Ahmad : A study of digital mammograms by using clustering algorithms. *Journal of Scientific and Industrial Research* 71.9 : 594 (2012).

[Hall et al. 2009] Hall M., Frank E., Holmes G., Pfahringer B., Reutemann P., and Witten I. H. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter* 11.1: 10-18 (2009).

[Haloi 2015] Haloi Mrinal. Improved microaneurysm detection using deep neural networks. *arXiv preprint arXiv*:1505.04424 (2015).

[Han et al.2006] Jiawei Han and Micheline Kamber. Data Mining: concepts and techniques 2nd edition. *Elsevier* (2006).

[Harris 2011] Papadopoulos Harris: Reliable probabilistic prediction for medical decision support. *Artificial Intelligence Applications and Innovations. Springer Berlin Heidelberg*, p.265-274 (2011).

[Harper 2005] Harper Paul R.: A review and comparison of classification algorithms for medical decision making. *Health Policy* 71.3: 315-331 (2005).

[Huang et al. 2005] Huang Jin and Charles X. Ling: Dynamic ensemble re-construction for better ranking. *European Conference on Principles of Data Mining and Knowledge Discovery.* pp.511-518 *Springer Berlin Heidelberg*, (2005).

[Huang et al. 2012] Mei-Ling Huang, Yung-Hsiang Hung, Wen-Ming Lee, R. K. Li, Tzu-Hao Wang: Usage of case-based reasoning, neural network and adaptive neuro-fuzzy inference system classification techniques in breast cancer dataset classification diagnosis. *Journal of medical systems* 36.2: 407-414 (2012).

[Hyontai 2012] Sug Hyontai: Better Decision Tree Induction for Limited Data Sets of Liver Disease. *Computer Applications for Bio-technology, Multimedia, and Ubiquitous City. Springer Berlin Heidelberg*, pp.88-93 (2012).

[Hyontai 2013] Sug Hyontai: Generating CART Decision Trees for Health Data Sets. *Technical Reports, University of Dongseo* (2013)

[Jacob 2012] Jacob Shomona Gracia, and R. Geetha Ramani : Mining of classification patterns in clinical data through data mining algorithms. *ICACCI* 2012: 997-1003 ACM, (2012).

[Jadhav et al.2011] Jadhav Shivajirao, Sanjay Nalbalwar, and Ashok Ghatol: Modular neural network model based foetal state classification. *International Conference on Bioinformatics and Biomedicine Workshops (BIBMW), p*p. 915-917 *IEEE* (2011).

[Jankisharan et al. 2014] Jankisharan Pahareeya, Rajan Vohra, and Jagdish Makhijani: Liver patient classification using intelligent techniques. *International Journal of Advanced Research in Computer Science and software Engg* 4.2: 295-299 (2014).

[Jinyan et al. 2015] Li Jinyan, S. Mohammed, Fiaidhi J. Improving the classification performance of biological imbalanced datasets by swarm optimization algorithms. *The Journal of Supercomputing 72.*10: 3708-3728 (2015).

[John et al. 1995] John George H., and Pat Langley. Estimating continuous distributions in Bayesian classifiers. *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc*. pp. 338-345, (1995).

[Jossinet 1996] Jossinet, J. : Variability of impedivity in normal and pathological breast tissue. *Medical and Biological Engineering and Computing* 34.5: 346-350 (1996).

[Kathleen et al. 2013] Miao Kathleen H., et al.: Mammographic Diagnosis for Breast Cancer Biopsy Predictions Using Neural Network Classification Model and Receiver Operating Characteristic (ROC) Curve Evaluation*. Journal of Selected Area in Bioinformatics (JBIO)* 3.9 (2013).

[Keerthi et al.2001] Keerthi S. S., Shevade S. K., Bhattacharyya C. and Murthy, K. R. K., Improvements to Platt's SMO algorithm for SVM classifier design. *Neural Computation* 13.3: 637-649 (2001).

[Kharya et al.2014] Kharya S., S. Agrawal, and S. Soni : Using Bayesian Belief Networks for Prognosis & Diagnosis of Breast Cancer. *International Journal of Advanced Research in Computer and Communication Engineering* 3.2 (2014).

[Kononenko 2001] Kononenko Igor. Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in medicine* 23.1: 89-109 (2001).

[Kurgan et al. 2001] Kurgan, L. A., Cios, K. J., Tadeusiewicz, R., Ogiela, M., & Goodenday, L. S. Knowledge discovery approach to automated cardiac SPECT diagnosis*. Artificial intelligence in medicine* 23.2: 149-169 (2001).

[Kuperman et al.2007] Kuperman G. J., Bobb A., Payne T. H., Avery A. J., Gandhi T. K., Burns G., and Bates D. W. Medication-related clinical decision support in computerized provider order entry systems: a review. *Journal of the American Medical Informatics Association* 14.1: 29-40 (2007).

[Kukar et al.2002] Kukar Matjaž, and Igor Kononenko. Reliable classifications with machine learning. *European Conference on Machine Learning*. Springer Berlin Heidelberg (2002).

[Kukar et al.2005] Kukar Matjaž, and Ciril Grošelj. Transductive machine learning for reliable medical diagnostics. *Journal of medical systems* 29.1: 13-32 (2005).

[Lavrac 1998] Lavrač Nada. Data mining in medicine: Selected techniques and applications. *In proceedings of intelligent data analysis in Medicine and Pharmacology* pp.11-31 (1998).

[Li et al., 2012] Jun-Bao Li, Yang Yu, Zhi-Ming Yang, Lin-Lin Tang: Breast Tissue Image Classification Based on Semi-supervised Locality Discriminant Projection with Kernels. *Journal of medical systems*, *36.*5: 2779-2786 (2012).

[Liang et al. 2013] Liang Chunlin, and Lingxi Peng: An automated diagnosis system of liver disease using artificial immune and genetic algorithms. *Journal of medical systems* 37.2: 1-10 (2013).

[Lianq et al. 2009] Liang Faming, and Jian Zhang. Learning Bayesian networks for discrete data. *Computational Statistics & Data Analysis* 53.4: 865-876 (2009).

[Linoff et al.2011] Linoff, Gordon S., and Michael JA Berry. Data mining techniques: for marketing, sales, and customer relationship management. *John Wiley & Sons* (2011).

[Ludwig 2010] Ludwig Simone A. : Prediction of breast cancer biopsy outcomes using a distributed genetic programming approach*. The 1st ACM International Health Informatics Symposium* pp. 694-699, (2010).

[M. & E. 2015] Mokhtar Sahar A., and Alaa Elsayad : Predicting the Severity of Breast Masses with Data Mining Methods. *arXiv preprint arXiv:*1305.7057 (2013).

[M. & J. 2010] Marques de Sá JP and Jossinet J : UCI machine learning repository, http://archive.ics.uci.edu/ml/datasets/ Breast+Tissue , (2013).

[Maciej 2014] Zięba, Maciej. Service-oriented medical system for supporting decisions with missing and imbalanced data*. IEEE journal of biomedical and health informatics* 18.5: 1533-1540 (2014).

[Maciej et al. 2013] Maciej Z, Tomczak JM, Lubicz M, Witek J Boosted SVM for extracting rules from imbalanced data in application to prediction of the post-operative life expectancy in the lung cancer patients. *Applied soft computing, vol* 14, *Elsevier*, pp 99–108, (2014).

[Maciej et al. 2013] Zięba, Maciej, and Jerzy Świątek. Ensemble SVM for imbalanced data and missing values in postoperative risk management. *IEEE 15th International Conference on e-Health Networking, Applications & Services (Healthcom),* pp. 95-99 (2013).

[Maddaen 2002] Madden Michael G. Evaluation of the performance of the Markov blanket bayesian classifier algorithm. *arXiv preprint cs*/0211003 (2002).

[Mane et al.2014] Mane Vijay M. and Dattatray V. Jadhav, Review: Progress Towards Automated Early Stage Detection of Diabetic Retinopathy: Image Analysis Systems and Potential*. Journal of Medical and Biological Engineering* 34.6: 520-527 (2014).

[Mazaheri et al.2015] Mazaheri, Parastoo, Anis Norouzi, and Abbas Karimi: Using algorithms to predict liver disease Classification. *Electronics Information & Planning* 3 (2015).

[McBride et al. 2014] McBride Ryan, Ke Wang and Wenyuan Li. Classification by CUT: Clearance Under Threshold. *International Conference on Data Mining,* pp. 410-419 *IEEE* (2014).

[McSherry 2011] McSherry David. Conversational case-based reasoning in medical decision making. *Artificial intelligence in medicine* 52.2: 59-66 (2011).

[Mitchell 1997] Mitchell Tom M. Machine learning. *WCB McGraw-Hill* (1997).

[Muhammad et al.2015] Muhammad Iqbal, and Zhu Yan. Supervised Machine Learning Approaches: A Survey. *ICTACT Journal on Soft Computing* 5.3 (2015).

[Museli et al.2007] Muselli Marco, and Francesca Ruffino. Reliable learning: a theoretical framework. *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer Berlin Heidelberg, pp. 174-183 (2007).

[Nonte 2013] Nonte M.: Classification of Breast Tissue Using Electrical Impedance Spectroscopy. *Technical Reports, University of WISCONSIN-MADISON*, (2013).

[Nachev et al. 2015] Nachev Anatoli, and T. Reapy. Predictive Models for Post-Operative Life Expectancy after Thoracic Surgery. *Mathematical and Software Engineering* 1.1: 1-5 (2015).

[Naeem, 2014] Naeem, M. Etiological Evaluation of Seminal Traits Using Bayesian Belief Network. *International Journal of Bio-Science and Bio-Technology* 6.6: 79-86 (2014).

[Ocak et al.2013] Ocak Hasan, and Huseyin Metin Ertunc. Prediction of fetal state from the cardiotocogram recordings using adaptive neuro-fuzzy inference systems. *Neural Computing and Applications* 23.6: 1583-1589 (2013).

[Ocak 2013] Ocak Hasan. A medical decision support system based on support vector machines and the genetic algorithm for the evaluation of fetal well-being. *Journal of medical systems* 37.2: 1-9 (2013).

[Oh et al. 2011] Sangyoon Oh, Min Su Lee, and Byoung-Tak Zhang: Ensemble learning with active example selection for imbalanced biomedical data classification. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)* 8.2: 316-325 (2011).

[Onisko et al. 2013] Oniśko Agnieszka and Marek J. Druzdzel: Impact of precision of Bayesian network parameters on accuracy of medical diagnostic systems. *Artificial intelligence in medicine* 57.3: 197-206 (2013).

[Parvez et al. 2015] Ahmad Parvez, Saqib Qamar, and Syed Qasim Afser Rizvi: Techniques of data mining in healthcare: A review. *International Journal of Computer Applications* 120.15 (2015).

[Pawel 2015] Cichosz Pawel: Data Mining Algorithms: Explained Using R. *John Wiley & Sons* (2015).

[Peng et al. 2010] Peng Yonghong, Zhiqing Wu, and Jianmin Jiang: A novel feature selection approach for biomedical data classification. *Journal of Biomedical Informatics* 43.1: 15-23 (2010).

[Platt 1998] Platt John. Sequential minimal optimization: A fast algorithm for training support vector machines (1998).

[Polat et al. 2009] Polat Kemal and Salih Güneş: A new feature selection method on classification of medical datasets: Kernel F-score feature selection. *Expert Systems with Applications* 36.7: 10367-10373 (2009).

[Quinlan 2014] Quinlan J. Ross, C4. 5: programs for machine learning. *Elsevier* (2014).

[R&B&V] Bendi Venkata Ramana, Prof. M. Surendra Prasad Babu and Prof. N. B. Venkateswarlu: UCI machine learning repository, https://archive.ics.uci.edu/ml/datasets/ILPD+%28Indian+Liver+Patient+Dataset%29 , (2012).

[Ramana et al.2011] Ramana Bendi Venkata, M. Surendra Prasad Babu, and N. B. Venkateswarlu. A critical evaluation of bayesian classifier for liver diagnosis using bagging and boosting methods. *International Journal of Engineering Science and Technology* 1.3: 2422-2426 (2011).

[Ramana et al.2012] Ramana, B. V., Babu, M. S. P., & Venkateswarlu, N. B.: A critical comparative study of liver patients from usa and india: An exploratory analysis. *International Journal of Computer Science Issues* 9.2: 506-516, (2012).

[Ramana et al.2012_2] Ramana Bendi Venkata, M. Surendra Prasad Babu and N. B. Venkateswarlu. Liver classification using modified rotation forest. *International Journal of Engineering Research and Development* 6.1: 17-24 (2012).

[Rensimer et al.2000] Rensimer Edward R., Jacqueline P. Tomsovic and Pamela A. Wright: System and method for recording patient history data about on-going physician care procedures. U.S. Patent No. 6,154,726. 28 Nov. (2000).

[Russell et al.2009] Russell Stuart, Norvig Peter. Artificial Intelligence: A Modern Approach (3rd ed. 2009). *Prentice Hall*. ISBN 978-0137903955.

[Samuel 1959] Samuel Arthur L.: Some studies in machine learning using the game of checkers. *IBM Journal of research and development* 3.3: 210-229 (1959).

[Schafer et al.2002] Schafer Joseph L. and John W. Graham: Missing data: our view of the state of the art. *Psychological methods* 7.2: 147 (2002).

[Sharaf et al. 2013] Sharaf-el Deen Dina A., Ibrahim F. Moawad, and M. E. Khalifa : A breast cancer diagnosis system using hybrid case-based approach*. International Journal of Computer Applications* 72.23 (2013).

[Silva et al.2000] Da Silva, J. E., De Sá, J. M., & Jossinet, J. : Classification of breast tissue by electrical impedance spectroscopy. *Medical and Biological Engineering and Computing* 38.1: 26-30, (2000).

[Sim et al.2001] Sim I., Gorman P., Greenes R. A., Haynes R. B., Kaplan B., Lehmann H. and Tang P. C.: Clinical decision support systems for the practice of evidence-based medicine. *Journal of the American Medical Informatics Association* 8.6: 527-534 (2001).

[Sindhu et al. 2014] Sindhu V., Prabha S., Veni S., Hemalatha M.,THORACIC SURGERY ANALYSIS USING DATA MINING TECHNIQUES*, Int.J.Computer Technology & Applications* 5.2: 578-586 (2014).

[Sittig et al.2008] Sittig Dean F., Wright A., Osheroff J. A., Middleton B., Teich J. M., Ash J. S. and Bates D. W.: Grand challenges in clinical decision support. *Journal of biomedical informatics* 41.2 (2008): 387-392.

[Sundar et al.2012] Sundar C., M. Chitradevi, and G. Geetharamani: Classification of cardiotocogram data using neural network based machine learning technique. *International Journal of Computer Applications* 47.14 (2012).

[Tiwari et al. 2013] Tiwari Anil Kumar, Lokesh Kumar Sharma, and G. Rama Krishna: Comparative Study of Artificial Neural Network based Classification for Liver Patient. *Journal of Information Engineering and Applications* 3.4 2225-0506 (2013).

[Tomas et al.2013] Tomáš P., Krohova J., Dohnalek P., & Gajdoš P: Classification of cardiotocography records by random forest. *36th International Conference on Telecommunications and Signal Processing (TSP)* pp. 620-923 *IEEE* (2013).

[V. & D. 2015] Vijayarani S., and S. Dhayanand: Liver disease prediction using SVM and Naïve Bayes algorithms. *International Journal of Science, Engineering and Technology Research (IJSETR) 4.*4: 816-820 (2015).

[Vapnik 1995] V. Vapnik, The nature of statistical learning theory. *Springer-Verlag: New York* (1995).

[Viera et al.2005] Viera Anthony J. and Joanne M. Garrett: Understanding interobserver agreement: the kappa statistic. *Fam Med* 37.5: 360-363 (2005).

[Vladimir et al.2005] Vovk Vladimir, Alex Gammerman, and Glenn Shafer: Algorithmic learning in a random *world. Springer Science & Business Media* (2005).

[Wang et al. 2014] Hong Wang, Qingsong Xu, Lifeng Zhou: Seminal Quality Prediction Using Clustering-Based Decision Forests. *Algorithms* 7.3: 405-417 (2014).

[Witten et al.2005] Witten Ian H., and Eibe Frank. Data Mining: Practical machine learning tools and techniques. *Morgan Kaufmann* (2005).

[Y. & H., 2011] Tingxu Yan, Yuexian Hou: An unsupervised feature selection method based on degree of feature cooperation. *Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)* 2:1300-1306 (2011).

[YiNan et al.2016] Zhang Yinan and Mingqiang An: An Active Learning Classifier for Further Reducing Diabetic Retinopathy Screening System Cost. *Computational and Mathematical Methods in Medicine* (2016).

[Yilmaz et al.2013] Yılmaz Ersen, and Çağlar Kılıkçıer: Determination of fetal state from cardiotocogram using LS-SVM with particle swarm optimization and binary decision tree. *Computational and mathematical methods in medicine* (2013).

[Yoo et al. 2012] Yoo, Illhoi, Patricia Alafaireet, Miroslav Marinov, Keila Pena-Hernandez, Rajitha Gopidi, Jia-Fu Chang, and Lei Hua: Data mining in healthcare and biomedicine: a survey of the literature. *Journal of medical systems* 36.4: 2431-2448 (2012).

[Zhang et al. 2015] Shu Zhang, Samira Sadaoui, Malek Mouhoub: An Empirical Analysis of Imbalanced Data Classification. *Computer and Information Science* 8.1: 151-162 (2015).