DEGREE PROJECT IN COMPUTER ENGINEERING,
FIRST CYCLE, 15 CREDITS
STOCKHOLM, SWEDEN 2020

Evaluation of Machine Learning classifiers for Breast Cancer Classification

ROBIN DANG

ANDERS NILSSON

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

TRITA -EECS-EX-2020:404

Evaluation of Machine Learning classifiers for Breast Cancer Classification

ROBIN DANG & ANDERS NILSSON

Degree project in Computer Science, 15 credits
Stockholm, Sweden
Date: June 1, 2020
Supervisor: Jeanette Hällgren Kotaleski
Examiner: Pawel Herman
Royal Institute of Technology KTH
School of Electrical Engineering and Computer Science
Swedish title: Evaluering av olika Maskininlärnings klassificerare för bröstcancerklassificering

Abstract

Breast cancer is a common and fatal disease among women globally, where early detection is vital to improve the prognosis of patients. In today's digital society, computers and complex algorithms can evaluate and diagnose diseases more efficiently and with greater certainty than experienced doctors. Several studies have been conducted to automate medical imaging techniques, by utilizing machine learning techniques, to predict and detect breast cancer. In this report, the suitability of using machine learning to classify whether breast cancer is of benign or malignant characteristic is evaluated. More specifically, five different machine learning methods are examined and compared. Furthermore, we investigate how the efficiency of the methods, with regards to classification accuracy and execution time, is affected by the preprocessing method Principal component analysis and the ensemble method Bootstrap aggregating. In theory, both methods should favor certain machine learning methods and consequently increase the classification accuracy. The study is based on a well-known breast cancer dataset from Wisconsin which is used to train the algorithms. The result was evaluated by applying statistical methods concerning the classification accuracy, sensitivity and execution time. Consequently, the results are then compared between the different classifiers.

The study showed that the use of neither Principal component analysis nor Bootstrap aggregating resulted in any significant improvements in classification accuracy. However, the results showed that the support vector machines classifiers were the better performer. As the survey was limited in terms of the amount of datasets and the choice of different evaluation methods with associating adjustments, it is uncertain whether the obtained result can be generalized over other datasets or populations.

Sammanfattning

Bröstcancer är en vanlig och dödlig sjukdom bland kvinnor globalt där en tidig upptäckt är avgörande för att förbättra prognosen för patienter. I dagens digitala samhälle kan datorer och komplexa algoritmer utvärdera och diagnostisera sjukdomar mer effektivt och med större säkerhet än erfarna läkare. Flera studier har genomförts för att automatisera tekniker med medicinska avbildningsmetoder, genom maskininlärnings tekniker, för att förutsäga och upptäcka bröstcancer. I den här rapport utvärderas och jämförs lämpligheten hos fem olika maskininlärningsmetoder att klassificera huruvida bröstcancer är av god- eller elakartad karaktär. Vidare undersöks hur metodernas effektivitet, med avseende på klassificeringssäkerhet samt exekveringstid, påverkas av förbehandlingsmetoden Principal component analysis samt ensemble metoden Bootstrap aggregating. I teorin skall båda förbehandlingsmetoder gynna vissa maskininlärningsmetoder och således öka klassifiseringssäkerheten. Undersökningen är baserat på ett välkänt bröstcancer dataset från Wisconsin som används till att träna algoritmerna. Resultaten är evaluerade genom applicering av statistiska metoder där träffsäkerhet, känslighet och exekveringstid tagits till hänsyn. Följaktligen jämförs resultaten mellan de olika klassificerarna.

Undersökningen visade att användningen av varken Principal component analysis eller Bootstrap aggregating resulterade i några nämnvärda förbättringar med avseende på klassificeringssäkerhet. Dock visade resultaten att klassificerarna Support vector machines Linear och RBF presterade bäst. I och med att undersökningen var begränsad med avseende på antalet dataset samt val av olika evalueringsmetoder med medförande justeringar är det därför osäkert huruvida det erhållna resultatet kan generaliseras över andra dataset och populationer.

Contents

Chapter 1

Introduction

1.1 Overview

Artificial intelligence is a field of computer science where learning is an extensive part to make computers more intelligent. In today's digital society, artificial intelligence has become more actualized and applicable in daily social functions as a result of increasing data production and storage. One major branch of artificial intelligence, and also one of the most rapidly growing development fields of artificial intelligence and computer science overall, is machine learning which serves as a tool for intelligent data-/predictive analysis utilizing complex algorithms [1].

The main concept regarding machine learning is to feed the algorithms with different kinds of data, mainly from generalized examples, to analyze and assess real-time data to perform complex tasks. With the large amount of data flowing in today's digital networks, manual programming is not an alternative in data analysis. Machine learning is oftentimes feasible and cost-effective as more data becomes available and more complex problems could be solved, which has made it widely used in different areas of computer science [1].

From the very beginning machine learning was designed to analyze medical datasets and to predict, detect, and estimate different diagnoses in patients [2]. The field of using machine learning in medical aid is often referred to as computer-aided diagnoses (CAD), which consists of systems to interpret medical images of e.g X-rays, MRI, and ultrasound.

The digital revolution has provided effective methods to collect and store data.

Today's modern hospitals are well equipped with data collection devices to gather and monitor data to be shared in information networks. The technology of machine learning is well suited to analyze such data and has proven to be very effective in diagnosing and detecting patterns and traces of diseases, and especially cancer, where the data mainly consists of parts of medical records [2].

Worldwide, breast cancer is the most common form of cancer in women and the second most common cancer overall. The survival rate varies greatly between developed and less developed countries where medical methods used to detect breast cancer differ in quality [3]. Several machine learning techniques have been introduced to the topic and are nowadays frequently used in breast cancer assessment [4].

Several studies have been conducted and proven that intelligent computer programs can diagnose breast cancer more efficiently and accurate than experienced human medical practitioners [5]. Then the naturally emerging question states: *"How suitable is the utilization of machine learning methods in diagnosing and classifying breast cancer and in fact, how effective is it?"*. As computers nowadays solves problems faster than humans makes this field of computer science very fascinating. Especially if it could benefit and contribute with quality assurance to healthcare and the medical treatment of breast cancer.

1.2 Motivation

Within healthcare, medical data analysis based on machine learning is a well-studied field with several research studies conducted. In recent times, several studies have been applied to computer-aided diagnosis (CAD), a field of medical analysis. CAD is a great value research topic as old-fashioned diagnostic methods are considered dull and imprecise. The usage of such techniques in medical areas has proven to be beneficial as such approaches could be considered to be more efficient and precise which is of great assistance in the decision making [5]. The increasing trend of breast cancer cases in the world motivates further medical research within the application of CAD and machine learning. As time is of the essence of breast cancer treatment, utilizing machine learning tools would most likely accelerate such process's to improve the quality of the medical treatment.

One of the key points and aspects of this research covers the effects of

preprocessing methods on classification accuracy. Preprocessing methods are beneficial for some machine learning methods, in theorym, and should thus increase the classification performance.

1.3 Research Question

The purpose of this research is to analyze the suitability of machine learning to assess and classify cancer. More specifically, this research will focus on breast cancer classification and the prediction of whether a given observation is of benign or malignant nature. The research questions states as following:

- Which of the following Machine learning methods:
  - Naive Bayes Classifier
  - Support Vector Machine (Linear)
  - Support Vector Machine (RBF)
  - K-Nearest neighbors
  - Decision Tree

  is most suitable for analyzing and assess whether a patient, diagnosed with breast cancer, has benign and malignant characteristics with respect to the classification performance?
- What are the impacts of Bootstrap aggregating (Bagging) and Principal component analysis (PCA) on the classification performance concerning the prediction accuracy and execution time?

1.3.1 Hypothesis

Bagging is expected to reduce the variance and improve the classification accuracy of classifiers suffering from high variance [6]. Fully grown Decision trees generally suffer from high variance and Bagging is therefore expected to reduce the variance as well as improve the classification accuracy of the classifier.

The Naive Bayes classifier relies on a features independence assumption. Hence, the preprocessing technique PCA is expected to improve the performance of the Naive Bayes classifier. Since PCA is a dimensionality reduction technique, it is expected to generate better-generalized classifiers with shorter execution

time.

K-Nearest Neighbors is by the nature of its implementation considered a lazy and slow method [7]. Hence, expected to have a longer execution time compared to the other methods. The ensemble method Bagging fits the data to several classifiers and combines the result. Since generating several classifiers is computationally expensive, Bagging is expected to significantly increase the execution time.

1.4 Scope

The scope of this research is limited by the available datasets and the few selections of supervised machine learning methods used. When evaluating the result, consideration must be taken into the limitation of the classifiers and the statistical methods used. Consideration must also be taken to the parameter settings of the classifiers used which were predefined to the default values of Sklearn [8]. Note that other more advanced methods and several datasets could have been considered to gain a better and more generalized result. Also, keep in mind that the extracted dataset was collected in one of many possible ways of collecting medical data for breast cancer research.

Chapter 2

Background

This chapter gives a brief explanation of the core subjects of the research. It is divided into five parts where the first two introduce breast cancer diagnosis and Computer-aided diagnosis (CAD). The remaining three parts gives a brief overview of machine learning, classifiers relevant for this research and the state of the art applications associated with the topic.

2.1 Breast cancer diagnosis

Breast cancer is a disease where cells in the breast continuously grow out of control. There exists several kinds of breast cancer depending on which cells in the breast developing cancer. But generally, all breast cancer tumors can be categorized into two main groups: (i) Benign and (ii) Malignant. Through blood and lymph vessels, cancer can spread outside the breast to other parts of the body [9]. For medical treatment of cancer, it is crucial to identify normal, benign or malignant tissue at an early stage.

Worldwide, breast cancer is the most common form of cancer in women and the second most common form of cancer overall. The disease was accounted for 2.1 million incidences and 0.6 million deaths in 2018 [3]. The rate of women diagnosed for breast cancer has not changed over the last decade but the death rate has declined over time. Although, it remains the second leading cause of cancer death among women [9]. At present, there are no effective ways to prevent breast cancer. However, efficient diagnosis in an early stage can increase the chance of full recovery thus enabling the medical treatment at an early phase [10].

Breast cancer screening is a tool used for early breast cancer detection. It can be helpful to find tissue abnormalities early and is usually performed before any signs or symptoms of the disease. Among the many tools available for screening, the most common techniques are mammograms, MRI or ultrasound [9].

A mammogram is an X-ray image of the breast used to detect early signs of breast cancer. It has proven to be an effective method in early breast cancer detection and has recently managed to reduce mortality rates [9]. The MRI technique uses magnets and radio waves to take images of the breast and is used alongside with mammograms to screen high-risk patients. Breast MRI's may appear as abnormal even though when there is no cancer.

Regular screenings and mammograms can contribute to early detection and lower the risk of breast cancer death. However, there can still be cases of false positive/negative test results due to the difficulty of interpreting the screening images correctly. This could be harmful as it could lead to more expensive tests, unnecessary biopsies which is time-consuming and may cause patient anxiety. Additional potential harm from screenings includes pain during the procedure and exposure of radiation from X-rays. These shortcomings could delay cancer detection and treatment process leading to repeated exposures for radiation from X-rays and unnecessary torment of the patient [9].

2.2 Computer-aided diagnosis (CAD)

Medical images contain a great amount of information that medical practitioners have to evaluate and analyze abnormalities in a short amount of time. Digital imaging techniques like MRI, X-ray and ultrasound could also be harmful to the human body if acquired with high energy, which is needed to provide images with high-quality results. Hence to avoid tissue damaging, images are taken in less energy which results in poorer image quality and unclear evidence [11]. Consequently, in such cases, it becomes difficult to interpret the image results.

CAD systems were introduced to take on the mentioned issues and are used to improve the quality of the image analysis. It provides an objective interpretation of mammogram screenings by increasing the image quality of mammograms result in a cost-effective way. The use of CAD systems requires samples of a diagnose used for learning which, in breast cancer classification, includes

labels and attributes of mammography scans.

To reduce the number of unnecessary tests, various computer-aided diagnosis systems (CAD) have been proposed to support medical practitioners in different decision making. The use of CAD systems increases the chances to identify abnormal tissue signs at an earlier stage that a human medical practitioner fails to find. The systems produce a second objective opinion of the interpretation of medical images. The advanced technology of CAD consists of interdisciplinary fields of artificial intelligence, computer vision and image processing where detecting tumors is a typical application [11]. Even though the system has been used in four decades the result outcomes are not completely reliable. The system can't substitute a human medical practitioner but instead plays a supportive role to significantly make radiologists better decision-makers.

2.3 Machine learning

Machine learning is an application of artificial intelligence and the science of getting computers to learn and act as humans do. The learning process comprises feeding the algorithms with observations or real-world interactions in order to find patterns in the data, to enable better future decisions based on the provided examples. The primary purpose is to allow computers to learn automatically without human assistance [12].

The methods of learning algorithms used in machine learning are often categorized as supervised or unsupervised [12].

- Supervised learning means that the data examples provided are labeled to tell the machine exactly what patterns to look for.
- Unsupervised learning is the opposite of supervised learning where the data has no labels. The machine looks for whatever patterns it can find.
- Reinforcement learning methods learns by trial and error in order to achieve a clear objective over time.

2.3.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a dimensional reduction technique aiming to find a low-dimensional representation of the dataset that contains as much as possible of the information of the original dataset. PCA summarizes a large set of correlated variables with a smaller number of representative variables called principal components that collectively explain most of the variability in the original set. In PCA, the amount of information refers to the proportion of variance explained per principal components. The variance explained takes a value between 0 and 1 [6].

Currently, there is no well-accepted way to determine the optimal number of principal components to use. This makes PCA more of an art than a science. However, one commonly used method is visualizing and examining a scree plot, searching for a point at which the proportion of variance explained drops off. This point of the graph is often referred to as an elbow in the graph [6].

2.3.2 Bootstrap aggregating (Bagging)

Bootstrap aggregating (Bagging) is an ensemble method, which combines several machine learning techniques into one predictive model. The goal of Bagging is to improve the prediction accuracy by reducing the variance of the statistical learning method used. Hence, Bagging is particularly useful for classifiers that suffer from high variance [6].

Bagging involves following three steps:

1. Generating a couple of bootstrapped datasets from the original dataset.
2. Fitting each dataset to a separate classifier.
3. Combining all classifiers to form the final model.

In the classification setting, there exist several useful approaches to combine the result from the individual classifiers. One useful technique is the majority vote, selecting the most commonly occurring majority class among the predictions [6].

2.4 Classifiers

This section gives a brief introduction to the different classifiers used throughout this report. It is divided into four subsections, one per classifier describing

the main characteristics of the classifier. In general, all classifiers belongs to the category of supervised learning and have been extensively used in similar CAD Breast cancer research [13]

2.4.1 Naive Bayes classifier

Despite its simplicity, the Naive Bayes classifier has proven to perform sufficiently in computer-aided diagnosis and many other complex real-world situations [14]. The classifier is based on the well known Bayes' theorem and relies on a feature independent assumption where all features are assumed to be conditionally independent. The definition of Bayes theorem is given in 2.1 and 2.2.

$$posterior = \frac{likelihood \times prior}{evidence} \tag{2.1}$$

$$p(C_k|x) = \frac{p(x|C_k) \times p(C_k)}{p(x)} \tag{2.2}$$

Where C_k represents the class k, whereas x represents a vector of features. The posterior represents the probability of the class k given values of the features x, the likelihood represents the probability of x given the class k, the prior represents the probability of the class k without considering the features x and the evidence represents the probability of the features x before considering the class k.

The learning process consists of calculating the likelihood and the prior probability for each class. However, the evidence could be evicted since it achieves the same value for each class. New instances are then classified into the class with the highest posterior probability.

2.4.2 Support Vector Machines

Support vector machines (SVM) are a generalization of the Maximal margin and Support vector classifiers. The forerunners MM and SVC are limited to separate classes with a linear decision boundary whereas the decision boundary of SVM can take any form. The flexibility of the decision boundary is determined by the kernel function. The most common kernels are the linear, the polynomial and the radial basis function (RBF), where the RBF kernel defines the most flexible decision boundary. SVM relies on support vectors, a subset of the dataset used to identify the decision boundary between classes [15].

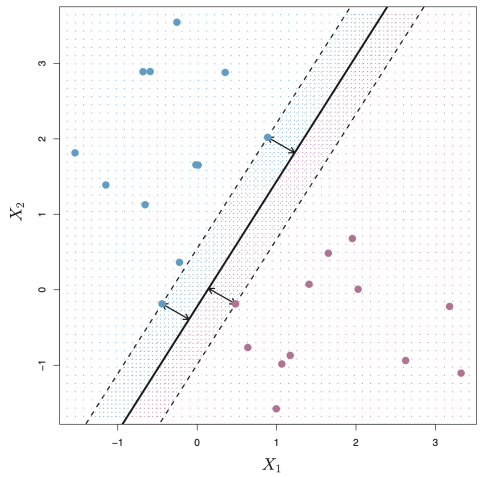


Figure 2.1: The decision boundary of a Support vector machine separating two classes in a two dimensional space. The support vectors appears on the dashed lines, called the margin.

Support vector machines are based on the assumption that almost all kinds of binary data become linearly separable when represented in a high-dimensional space. Since increasing the dimensionality is computationally expensive, SVM utilizes a process called the kernel trick. The kernel trick calculates the scalar product of the low-dimensional input data, creating the virtual transformation into a higher dimension [6].

2.4.3 K-Nearest Neighbors (KNN)

The K-Nearest Neighbors (KNN) classifier assumes that similar things are in close connection. This method captures the idea of similarity by implementing and applying pattern classification. The major task is to search for structures in the dataset and partition it into different categories. In practice this is commonly performed by calculating the Euclidean distance between the observation and its K nearest nodes of training data. The heavy calculation nature of the algorithm makes it slow for increasing volumes of input data [7]. However, it has proven to generate classifiers surprisingly close to the optimal Bayes classifier [6].

Formally, the KNN classifier attempts to estimate the conditional distribution of Y given X. The conditional probability is estimated for each class as the fraction of the K closest nodes N_0 whose response values belongs to the class.

The definition is given in formula 2.3. The KNN classifier proceeds by applying Bayes rule and classifies the given observation x_0 to the class with the greatest probability.

$$Pr(Y = j|X = x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i = j) \tag{2.3}$$

Where $I(y_i = j)$ is the indicator function, which equals 1 if $y_i = j$ and 0 otherwise. The choice of K is made by the user and might have a drastic effect on the prediction accuracy. A high value of K produces a decision boundary close to linear whereas a low value a classifier with huge flexibility.

2.4.4 Decision Trees

Decision trees utilize tree structures in pattern recognition to make predictions about observations. The leaves in the tree structure represent the predicted class labels whereas the branches correspond to conjunctions of features. In the classification setting, several methods exist for labeling the leaves. A common method is to label a leaf with the value of the most occurring class.

Decision trees utilizes the Recursive binary splitting algorithm to grow the tree. The algorithm starts at the top of the tree and successively splits the observations associated with the current node. At each split, two new branches are created further down the tree. This process proceeds until a stopping criterion is reached. There are several measurements determining the optimal decision at each split. The most common is the Gini index which measures the total variance across the K classes. A formal definition of the Gini index is given in formula 2.4. Where p_{mk} refers to the proportion of training observations in the *m:th* node that belongs to the *k:th* class [6].

$$G = \sum_{k=1}^{K} \hat{p}_{mk} * (1 - \hat{p}_{mk}) \tag{2.4}$$

2.5 State of the art

In today's digital society, the developmental process in the field of machine learning is constantly making progress where technology is applied in various daily social functions. Within medical treatment, machine learning is a topical topic and appears in several contexts. Among these, the most common are

Computer Vision, Natural Language Processing and Robotics. Examples of current healthcare applications in each area:

- **Computer Vision:** Covers analysis of medical image segmentation/registration and Electroencephalography (EEG) [2].
- **Natural Language Processing (NLP):** Psychological/cognitive research and sentiment analysis [16].
- **Robotics:** Current applications in medical robotics deals with automation of delicate and non-invasive surgeries [17].

For any machine learning systems to be successful and useful in solving complex tasks, the following features are desired: good performance, ability to appropriately deal with missing and messy data and the ability to explain the decision making [2].

Our research applies to the field of Computer Vision, in the area of assessment and analysis of medical images such as X-rays and mammography screenings. Common techniques and approaches includes analysis of different datasets and/or interpretation of medical images. A state of the art research in breast cancer classification was conducted by *Ahmed Osmanovic et. al.*. The data used was collected from medical images describing the different characteristics of cells and then classified by using Neural network classifiers [18]. The obtained results were compared with the diagnosis given by a human medical practitioner. The research and methods of *Osmanovic's et. al.* is a state of the art approach, commonly used in the research of machine learning classification of breast cancer.

Another similar research paper, written by *Dumitru, Diana.* investigates the potential of using the Naive Bayesian classification methodology to predict recurrent events of breast cancer. This research utilizes the well-known Wisconsin prognostic breast cancer dataset and evaluates the classification efficiency with the measurements classification error rate, sensitivity and specificity. The findings of this paper show that the classifier provides performances equivalent to some of the highest results obtained by other classifiers in the field [14].

Chapter 3

Methods

This chapter deals with method selection and the procedures to implement the solution based on the provided dataset and literature research. It is divided into three parts. The first part describes the provided dataset. The second part gives the reader an insight into the implementation. The third part consists of a description of the measurements used to evaluate the result.

3.1 Wisconsin Breast Cancer dataset

This research is based on the Wisconsin Breast Cancer dataset. This dataset is intended to be used for classification purposes with the aim of training classifiers to be able to predict and assess future instances. The main characteristics of the dataset are summarized in table 3.1. The Wisconsin dataset has been frequently used in similar medical researches over the years which motivates the usage.

dataset	Instances	Benign	Malignant	Attributes	Characteristics
Wisconsin breast cancer dataset	569	357	212	32	Continuous

Table 3.1: A summary of the dataset, containing the number of instances and attributes, the characteristics of the attributes and the ratio between benign and malignant instances.

The dataset consists of 32 attributes where the first two are ID and diagnosis. Consequently, the remaining 30 attributes are used for the analysis which consists of different continuous features with four significant digits. The features are describing the characteristics of the cell nuclei and were computed

from a digitized image of a fine-needle aspirate (FNA) of breast lumps collected from 569 patients.

3.2 Implementation

Test methods were created to analyze the performance of the classifiers considering different adjustments of the input data. The implementation was written in Python, which covers various state of the art libraries with extensible features and predefined methods and tools for machine learning and statistical analysis. In this research, the Python library Scikit [8] was used for the different implementations of classifiers while the library Matplotlib [19] was used to visualize the results.

The classification was performed on the Wisconsin breast cancer dataset with five different types of classifiers described in section 2.4. Note that the SVM classifier was initialized twice, the first with a linear kernel and the second with an RBF kernel. The classifiers were also individually combined using the Bagging technique, resulting in another five classifiers. The data were normalized and randomly split into a fraction of 70%/30% on each run, according to the most common conventions to sample subsets of data [20].

3.2.1 Classifiers and parameters

All of the classifiers have several adjustable parameters but as performance measuring is the intention of this research, the parameters were set to the default values of Scikits internal implementation. The parameter values used are summarized in Appendix A. However, two different types of SVM classifier was used where the kernels were set to Linear and RBF as this generates a significant difference in the flexibility of the classifier.

To ensure fair comparisons and classifications throughout the research, global common parameters such as the amount of runs and fractions of training and test data were kept consistent between runs. All tests were executed 100 times per classifier, providing a strong and reliable result with feasible execution time. A greater amount of runs would significantly increase the execution time and at the same time not substantially improve the quality of the results.

3.3 Evaluation

This section briefly describes the metrics and measurements used to evaluate the performance of the classifiers. To generate a more trustworthy estimate and more reliable measurements, the metrics were calculated by averaging the results over 100 executions according to 3.2.1 [6]. Note that conclusions based on the different measurements alone are not enough estimates to make reasonable conclusions but if evaluated together good estimations can be made.

To evaluate the impact of utilizing PCA and Bagging techniques, comparisons were computed with regards to the unprocessed input data. By calculating the mean and visualizing the spread for all measures, the impact and the behavior of the input adjustments could be analyzed.

The classification accuracy, execution time, sensitivity and specificity are important reference points that were measured and described in more detail below. The aim was to find the most suitable classifier to distinguish between benign and malignant breast cancer. Although specificity and sensitivity are important factors, focus has been on classification accuracy as it technically covers them all.

3.3.1 Classification accuracy

Classification accuracy is an evaluation measurement that shows the fraction of correctly classified observations divided by the total amount of observations [6]. A more formal definition of classification accuracy is given in formula 3.1.

$$Accuracy = \frac{Amount\ of\ correct\ predictions}{Total\ amount\ of\ predictions} \tag{3.1}$$

With respect to binary classifications, accuracy can also be defined in terms of positives and negatives:

$$Accuracy = \frac{True\ positives + True\ negatives}{Total\ amount\ of\ predictions} \tag{3.2}$$

where

$$\begin{aligned} Total\ amount\ of\ predictions = True\ positives + True\ negatives \\ + False\ positives + False\ negatives \end{aligned} \tag{3.3}$$

3.3.2 Sensitivity

Sensitivity measures the proportion of the positive samples that were correctly classified [21]. The sensitivity is often referred to as the true positive rate. A formal definition is given in formula 3.4.

$$Sensitivity = \frac{True\ positive}{True\ positive + False\ negative} \tag{3.4}$$

In this report, positive samples refer to cancer of malignant characteristics and negative to benign characteristics. A reformulated formula is given in 3.5.

$$Sensitivity = \frac{Correct\ malign\ predictions}{Total\ amount\ of\ malign\ predictions} \tag{3.5}$$

3.3.3 Specificity

The specificity measures the proportion of negative samples that were correctly classified [21]. The specificity is closely related to sensitivity and often referred to as the true negative rate. A formal definition is given in formula 3.6.

$$Specificity = \frac{True\ negative}{False\ positive + True\ negative} \tag{3.6}$$

Applied to this report, the definition can be reformulated as in 3.7.

$$Specificity = \frac{Correct\ benign\ predictions}{Total\ amount\ of\ benign\ predictions} \tag{3.7}$$

3.3.4 Execution time

Since time is an important factor to consider during the process of breast cancer diagnosis and medical breast cancer treatment, the execution time was measured during each test. The execution time was calculated by averaging the execution time over 100 executions. The purpose was to analyze and assess if there was any significant difference in execution time between the classifiers and how the execution time was affected by PCA and Bagging. Note that the time performing PCA was not included. Hence, the outcome of the execution time measurements considering PCA should be interpreted with caution.

Chapter 4

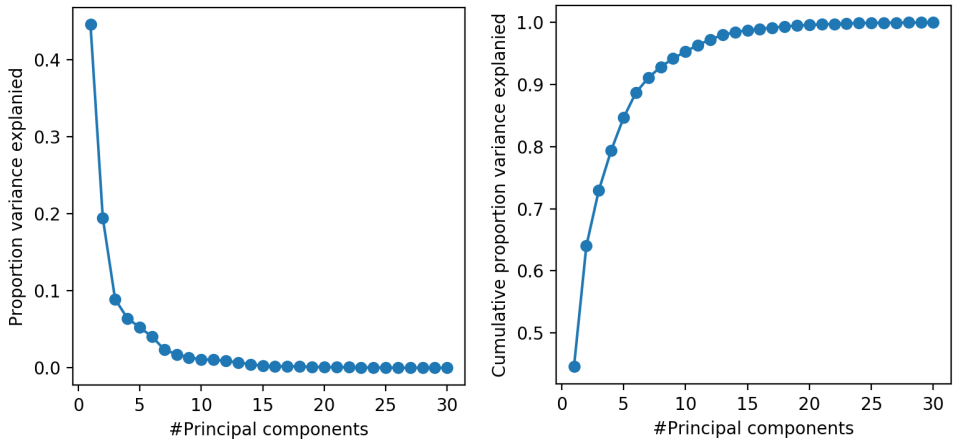
Results and Analysis

This section is divided into four parts. The first part contains the results of the PCA performed. The second part includes the results associated with the second research question, concerning the impact of utilizing PCA and Bagging. The third part contains the results associated with the first research question, regarding the comparison of classifiers. This part is based on the same data as in the previous section, but in a rearranged order. The last part deals with a brief summarized analysis of the results highlighting the most deviant observations. In sections 4.2 and 4.3, where the impact of using PCA and Bagging are conducted, the comparisons are made against a reference value denoted as **Naive**. In this context, **Naive** refers to the classifiers that neither utilizes PCA nor Bagging.

4.1 PCA

Figure 4.1 and table 4.1 represents the variance explained per principal component. As described in section 2.3.1, the optimal number of principal components to use is commonly determined by examining a scree plot, looking for a point representing an elbow.

The scree plot in 4.1 indicates that the use of seven principal components is the most suitable for this specific dataset and will be used as the reference value in further analysis. Table 3.1 also indicates that the 7:th principal component value is the most suitable by the asymptotic breakpoint where the decreasing rate of proportion variance and growth rate of the cumulative proportion variance is minimal which results in an elbow shape seen in figure 4.1.



(a) Proportion of variance explained (b) Cumulative representation

Figure 4.1: Scree plots visualizing the proportion of variance explained for the number of principal components. The left figure represents the proportion of variance explained per number of principal components. The figure to the right represents the cumulative proportion of variance explained per number of principal components.

# Principal components	1	2	3	4	5	6	**7**	8	9	10
Proportion var. explained	0.45	0.20	0.09	0.06	0.05	0.04	**0.02**	0.02	0.01	0.01
Cum. prop. var. explained	0.45	0.65	0.74	0.80	0.85	0.89	**0.91**	0.93	0.94	0.95

Table 4.1: Represents the proportion of variance explained per number of principal components. The highlighted value refers to the reference principal component value used in further analysis.

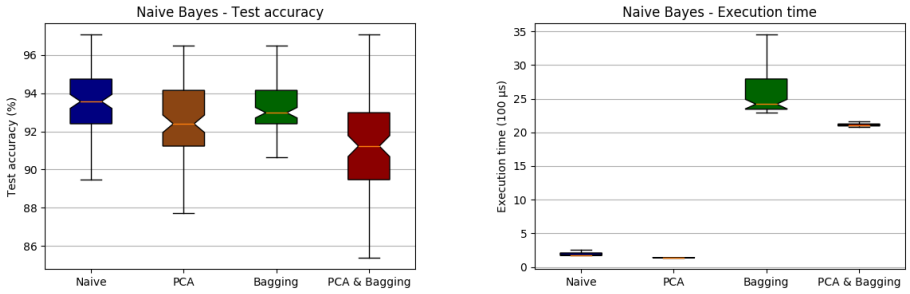
4.2 The impact of utilizing PCA & Bagging

The following subsections presents the impact on the classifiers by utilizing the PCA and Bagging methods. To gain an understanding about the effects of the dimensionality reduction of PCA and the assembling effects of Bagging on different classifiers, measurements were plotted in box plots and summarized in tables. The Naive preprocessing method was used as a reference value when comparisons were made. The box plot covers the distribution over 100 executions categorized by the method. The table presents the measurements, which composes of the five distinct measurements covered in 3.3, for each of the five classifiers. The values were generated by calculating the means of 100 executions for each classifier.

4.2.1 Naive Bayes

As observed in figure 4.2 and table 4.2, the classifier utilizing PCA performed slightly worse in each of the measurements except for the execution time. The Bagging version resulted in a significant increase of the execution time. However, Bagging also resulted in less spread of the test accuracy. Furthermore, the mean and median test accuracy decreased slightly.

The classifier utilizing PCA and Bagging had a significantly worse execution time than the reference value. As can be seen in table 4.2, the classifier also produced less satisfactory results in terms of specificity and classification accuracy. In 4.2, it is also noticeable that the variance of the test accuracy fluctuates to a greater extent for this classifier.



(a) Naive Bayes - Test accuracy

(b) Naive Bayes - Execution time

Figure 4.2: Box plots describing the accuracy and execution times of the Naive Bayes classifier categorized by the preprocessing method.

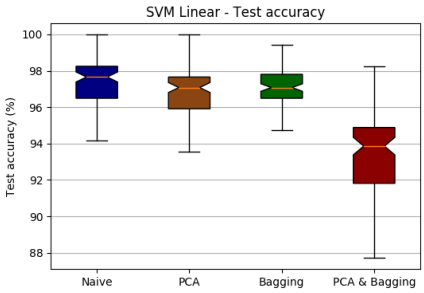
Test	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive	2.15	92.15 %	94.1 %	93.39 %	93.97 %
PCA	1.41	92.02 %	92.95 %	92.61 %	92.56 %
Bagging	26.67	91.51 %	94.18 %	93.2 %	93.76 %
PCA & Bagging	21.6	92.33 %	90.81 %	91.32 %	91.79 %

Table 4.2: Mean values of the measurements on 100 executions on the dataset with the Naive Bayes classifier, categorized by the preprocessing method.

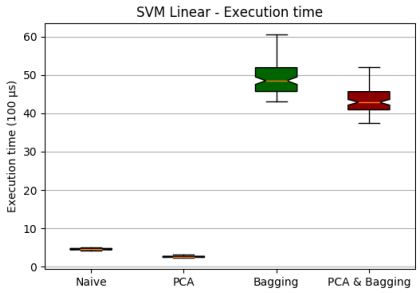
4.2.2 SVM Linear

From the observations of the graph in figure 4.3 and table 4.3, it is noticeable that the classifier utilizing PCA gained a slight increase of performance concerning the execution time. Consequently, this classifier had a slight deterioration in accuracy performance overall. Bagging resulted in a decreased variance of test accuracy. However, with the cost of a minor decrease of the mean and median test accuracy and a significant increase of the execution time.

Considering the classifier utilizing PCA and Bagging, the overall performance was worse than the reference value. The classifier had a major increase in execution time, a significant decrease in classification accuracy and a wider variance scope. As noticeable in table 4.3, the combined test resulted in a major deterioration in specificity in comparison to the other classifiers.



(a) SVM Linear - Test accuracy



(b) SVM Linear - Execution time

Figure 4.3: Box plots describing the accuracy and execution times of the SVM Linear classifier categorized by the preprocessing method.

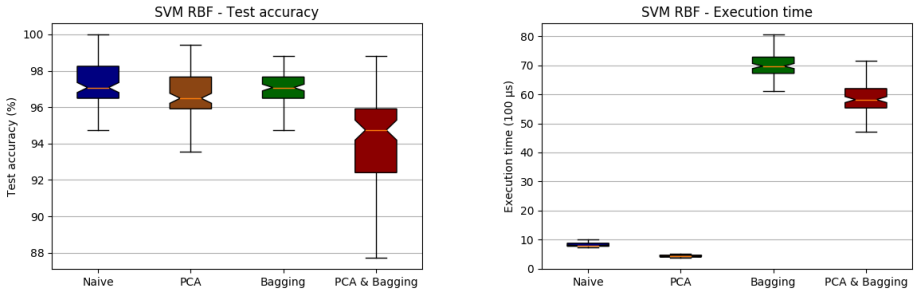
Test	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive	4.93	99.12 %	96.22 %	97.25 %	97.87 %
PCA	2.8	98.68 %	95.73 %	96.78 %	97.35 %
Bagging	50.28	98.74 %	96.12 %	97.05 %	97.67 %
PCA & Bagging	44.45	97.14 %	87.71 %	90.46 %	90.93 %

Table 4.3: Mean values of the measurements on 100 executions on the dataset with the SVM Linear classifier, categorized by the preprocessing method.

4.2.3 SVM RBF

Proceeding on the graph of figure 4.4 and table 4.4, the classifier utilizing PCA generated a wider spread of classification accuracy. However, the execution time added up to approximately half of the reference value. The overall accuracy performance had a slight decrease in comparison to the other classifiers. Concerning the effects of Bagging one can observe a notably increase in execution time, a minor decrease of classification accuracy and a reduced variance of classification accuracy between runs.

The classifier utilizing PCA and Bagging contributed to a significant performance decrease where the overall classification accuracy was slightly worse and with a wider variance.



(a) SVM RBF - Test accuracy

(b) SVM RBF - Execution time

Figure 4.4: Box plots describing the accuracy and execution times of the SVM RBF classifier categorized by the preprocessing method.

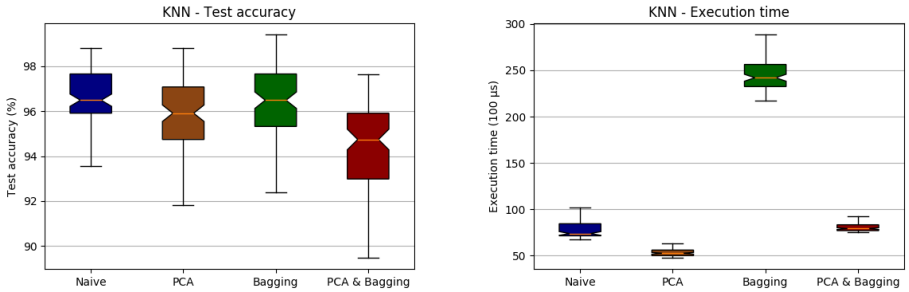
Test	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive	8.69	97.22 %	97.35 %	97.3 %	98.67 %
PCA	4.63	96.45 %	96.76 %	96.65 %	97.89 %
Bagging	70.46	96.67 %	97.17 %	96.98 %	98.47 %
PCA & Bagging	60.38	94.97 %	92.67 %	93.45 %	95.14 %

Table 4.4: Mean values of the measurements on 100 executions on the dataset with the SVM RBF classifier, categorized by the preprocessing method.

4.2.4 K-Nearest Neighbor

By inspecting figure 4.4 and table 4.4 it can be observed that the classifier utilizing PCA resultet in a broader classification accuracy variance and a minor decrease of classification accuracy on average. However, this classifier had the best execution time. Considering the classifier utilizing Bagging, one can observe a wider spread of classification accuracy. The execution time notably increased and the classification accuracy decreased in general.

The classifier utilizing both PCA and Bagging resulted in much better execution time in comparison to the Bagging version. Consequently, this classifier also resulted in a wider classification accuracy spread between runs and a deterioration of the classification accuracy on average.



(a) KNN - Test accuracy

(b) KNN - Execution time

Figure 4.5: Box plots describing the accuracy and execution times of the KNN classifier categorized by the preprocessing method.

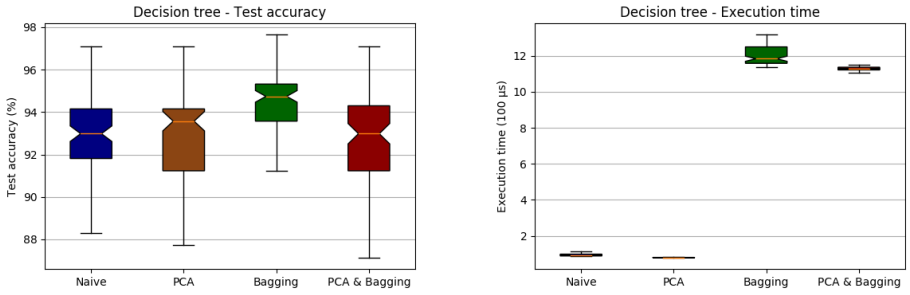
Test	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive	80.87	98.07 %	95.79 %	96.6 %	97.47 %
PCA	53.54	96.72 %	95.36 %	95.85 %	97.2 %
Bagging	248.07	97.95 %	95.63 %	96.45 %	97.74 %
PCA & Bagging	84.08	96.1 %	92.93 %	94.0 %	97.28 %

Table 4.5: Mean values of the measurements on 100 executions on the dataset with the KNN classifier, categorized by the preprocessing method.

4.2.5 Decision tree

Observed from the graph in figure 4.6 and table 4.6, the classifier utilizing PCA performed better considering execution time, sensitivity and classification accuracy. Although, this classifier had a slight decrease in specificity.

Regarding the classifier utilizing Bagging, the improved classification accuracy is noticeable. This classifier also resulted in a more narrow test accuracy spread. However, the execution time increased significantly for this classifier. Considering the combined version, there was a slight improvement in specificity but a deterioration in the remaining measurements.



(a) Decision tree - Test accuracy

(b) Decision tree - Execution time

Figure 4.6: Box plots describing the accuracy and execution times of the Decision tree classifier categorized by the preprocessing method.

Test	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive	1.03	89.54 %	94.47 %	92.59 %	100.0 %
PCA	0.86	90.56 %	94.15 %	92.8 %	100.0 %
Bagging	12.56	92.16 %	96.32 %	94.74 %	99.71 %
PCA & Bagging	11.3	89.46 %	94.54 %	92.63 %	99.8 %

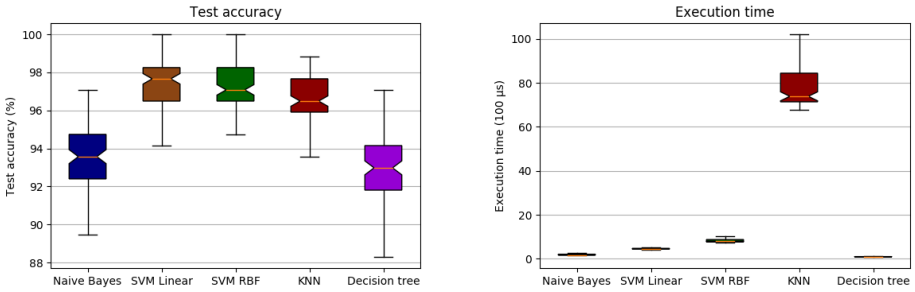
Table 4.6: Mean values of the measurements on 100 executions on the dataset with the Decision tree classifier, categorized by the preprocessing method.

4.3 Comparison of classifiers

The following subsections presents and visualises the comparisons between the five distinct classifiers by the different preprocessing method of the data input. To simplify the analysis and comparisons between the classifiers, two box plots and a table are presented in the same way as in section 4.2 but instead categorized by the five distinct classifiers for each preprocessing method. The box plot is generated by calculating the distribution over 100 executions on the dataset for each classifier while the measurement values of the table are calculated by a means of 100 executions categorized by the classifier.

4.3.1 Naive input

As can be seen in the graph of figure 4.7a, both SVM classifiers were the better performer in terms of classification accuracy and variance, where the in-between differences were negligible. Considering the classification accuracy and its spread between runs, one can also note a slightly worse performance by the Naive Bayes and the Decision tree classifiers. An explanation might be the substantial difference in mean sensitivity, which stands out in table 4.7. Figure 4.7b and table 4.7 points out the major differences in execution time where the KNN classifier is significantly slower than the remaining four classifiers.



(a) Test accuracy

(b) Execution time

Figure 4.7: Box plots describing the accuracy and executions times on a naive preprocessing of input data categorized by classifier.

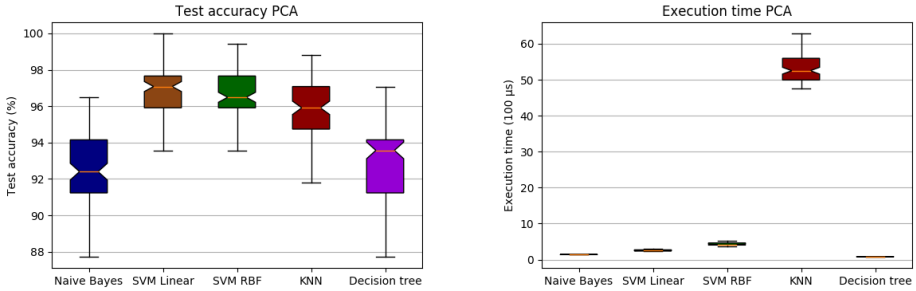
As observed in table 4.7, it can be noted that all of the five distinct classifiers had an average test accuracy over 90%. However, both of the SVM classifiers were the better performer averaging over 97% each in classification accuracy with a reasonable execution time.

Classifiers	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive Bayes	2.15	92.15 %	94.1 %	93.39 %	93.97 %
SVM Linear	4.93	99.12 %	96.22 %	97.25 %	97.87 %
SVM RBF	8.69	97.22 %	97.35 %	97.3 %	98.67 %
KNN	80.87	98.07 %	95.79 %	96.6 %	97.47 %
Decision tree	1.03	89.54 %	94.47 %	92.59 %	100.0 %

Table 4.7: Mean values of the measurements on 100 executions on the dataset with a naive preprocessing approach categorized by classifier.

4.3.2 PCA

It can be observed in the graph of figure 4.8 and table 4.8, that the PCA test follows a similar pattern from the results of section 4.3.1 where the behavior and performance of the five distinct classifiers acts correspondingly. As in section 4.3.1, the SVM classifier was the best performer considering classification accuracy. Considering the execution time, it is noticeable that the KNN classifier was the worst performer.



(a) Test accuracy PCA

(b) Execution time PCA

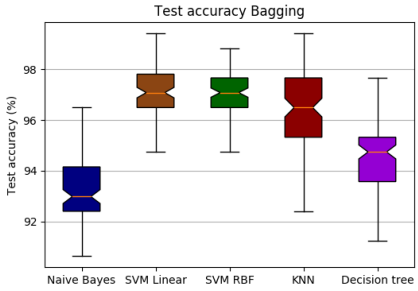
Figure 4.8: Box plots describing the accuracy and executions times on the PCA preprocessing of input data categorized by classifier.

Classifiers	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive Bayes	1.41	92.02 %	92.95 %	92.61 %	92.56 %
SVM Linear	2.8	98.68 %	95.73 %	96.78 %	97.35 %
SVM RBF	4.63	96.45 %	96.76 %	96.65 %	97.89 %
KNN	53.54	96.72 %	95.36 %	95.85 %	97.2 %
Decision tree	0.86	90.56 %	94.15 %	92.8 %	100.0 %

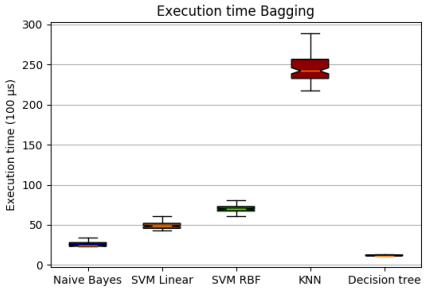
Table 4.8: Mean values of the measurements on 100 executions on the dataset with a PCA preprocessing approach categorized by classifier.

4.3.3 Bagging

Unlike the results from the previous two sections, it can be observed from figure 4.9a that the KNN classifier had it's peak value of classification accuracy. However, the KNN classifier also resulted in a much wider spread of the classification accuracy between the runs in comparison to SVM Linear and RBF classifiers. The wide classification accuracy variance of the KNN classifier had a significant impact on the median and mean classification accuracy which resulted in a decrease of the mean and median accuracy value. The results indicates that the SVM linear and RBF classifiers are more reliable in terms of the classification accuracy. Concerning classification accuracy, the Decision tree and the Naive Bayes classifiers performed the worst.



(a) Test accuracy Bagging



(b) Execution time Bagging

Figure 4.9: Box plots describing the accuracy and executions times on a Bagging preprocessing of input data categorized by classifier.

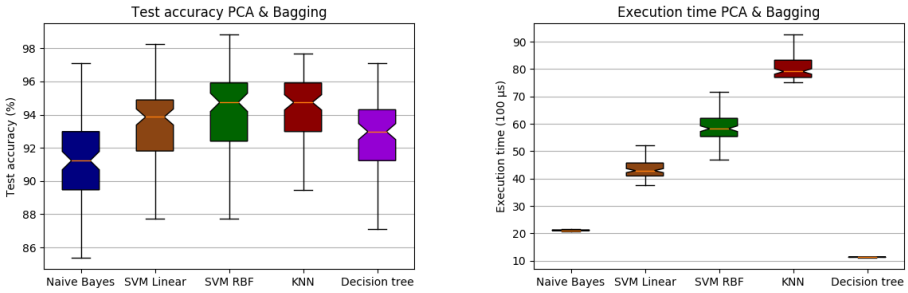
In accordance with previous results, it can be observed in figure 4.9b and table 4.9 that the KNN classifier resulted in an extensive execution time. The remaining classifiers held similar and stable results in terms of execution time, where the Decision tree and Naive Bayes classifiers performed slightly better.

Classifiers	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive Bayes	26.67	91.51 %	94.18 %	93.2 %	93.76 %
SVM Linear	50.28	98.74 %	96.12 %	97.05 %	97.67 %
SVM RBF	70.46	96.67 %	97.17 %	96.98 %	98.47 %
KNN	248.07	97.95 %	95.63 %	96.45 %	97.74 %
Decision tree	12.56	92.16 %	96.32 %	94.74 %	99.71 %

Table 4.9: Mean values of the measurements on 100 executions on the dataset with a Bagging preprocessing approach categorized by classifier.

4.3.4 PCA & Bagging

Considering the classifiers utilizing PCA and Bagging, proceeding on figure 4.10 and table 4.10 it is clearly noticeable that the distribution of classification accuracy and variance of all the five classifiers significantly increased. It is also observable that the overall performance was worse in general compared to the classifiers mentioned in previous sections.



(a) Test accuracy PCA & Bagging

(b) Execution time PCA & Bagging

Figure 4.10: Box plots describing the accuracy and executions times on a PCA and Bagging combined preprocessing of input data categorized by classifier.

Classifiers	Execution time (100 µs)	Sensitivity	Specificity	Test accuracy	Training accuracy
Naive Bayes	21.6	92.33 %	90.81 %	91.32 %	91.79 %
SVM Linear	44.45	97.14 %	87.71 %	90.46 %	90.93 %
SVM RBF	60.38	94.97 %	92.67 %	93.45 %	95.14 %
KNN	84.08	96.1 %	92.93 %	94.0 %	97.28 %
Decision tree	11.3	89.46 %	94.54 %	92.63 %	99.8 %

Table 4.10: Mean values of the measurements on 100 executions on the dataset with a combined PCA and Bagging preprocessing approach categorized by classifier.

4.4 Summary of results

The performance and behavior of the classifiers were distinguishable through all test cases. However, the results did not prove any significant difference in terms of performance by using PCA and Bagging except for the Decision tree classifier where the effects of Bagging had a positive impact on the classification accuracy. Especially the combination of PCA and Bagging generated a high fluctuation of classification accuracy. According to the results in section 4.2, all classifiers but the Decision tree classifier resulted in a decreased classification accuracy when applying PCA and Bagging. Regarding the execution time, it is evident that the classifiers utilizing Bagging were the least effective. Furthermore, the classifiers using PCA fairly decreased the execution time compared to the Naive preprocessing.

By the observations from section 4.3, the Naive and Bagging preprocessing methods generated the highest mean classification accuracy overall. It is also clear that the KNN classifier had the longest execution times among all test cases compared to the other classifiers. The performance of the two SVM classifiers stands out from the rest of the classifiers. Their classification accuracy were stable throughout the tests and with a reasonable execution time. The classifiers utilizing PCA and Bagging were distinctive from the rest of the tests where the performance decreased for all classifiers in terms of classification accuracy with fluctuating variance. This also impacted the execution time by significantly increasing the average runtime.

This could be explained by the reduced number of dimensions used, which significantly reduces the number of computations in the case of the KNN classifier.

Chapter 5

Discussion

Tests and evaluations were conducted to compare the performance and behavior of the classifiers, under the effects of different techniques, in classification of benign or malignant breast cancer tumours. The classification accuracy and execution time was measured and compared between different methods and classifiers used as stated in section 4.

5.1 The impact of PCA

It is interesting that PCA did not improve the classification accuracy for any of the classifiers. As presented in the result section 4, PCA generated a decreased mean accuracy with a wider spread between runs. It is also somewhat surprising that PCA did not increase the performance of the Naive Bayes classifier. Since Naive Bayes classifiers are based on the underlying assumption that each feature is independent, reducing the number of dimensions used is therefore expected to improve the performance. However, according to the results, this was not the case for this research. A possible reason for this is the fixed number of principal components used. Perhaps using different numbers of principal components would have generated a result accordingly to theory and our hypothesis. Another possibility is that the correlation between features in the dataset was low, resulting in an already satisfying classification accuracy without applying the preprocessing method.

The results indicated a slight improvement of execution time using PCA. However, it should be noted that when measuring the execution time for the classifiers using PCA, the execution time for the test itself was excluded. Hence, it is debatable whether the results indicating a slight improvement really is an

improvement.

5.2 The impact of Bagging

As stated in the hypothesis 1.3.1, Bagging is expected to increase the performance of classifiers that suffers from high variance. More specifically, Bagging aims to reduce the classification variance and is expected to be most beneficial for the Decision tree classifier that suffers from high variance by default. According to the result chapter 4, the hypothesis is consistent with the generated results, where the classification accuracy was improved for the Decision tree classifier. As for the remaining classifiers, there was no significant difference in classification accuracy.

Bagging also resulted in a more narrow spread of the variance. This was expected since the Bagging technique is used for reducing the variance of the classifier. However, it is interesting that utilizing Bagging generally led to a decreased classification accuracy. The explanation is probably due to the bias-variance trade-off. When aiming to reduce the variance, the bias is generally increased. If the increase in bias is greater than the decrease in variance, the classifier will performance worse considering classification accuracy.

5.3 PCA & Bagging combined

The combined version, using both PCA and Bagging, resulted in the highest fluctuating classification variance among all classifiers where the classification accuracy greatly decreased compared to PCA and Bagging in separate. A potential reason is that PCA reduces the number of dimensions used, which creates a constraint on the variability of used features for the Bagging technique, to gain an optimized generalization. As mentioned in section 2.3.2, Bagging aims to reduce the variance of the classifier. However, when the variability is limited, the variance to reduce is somewhat limited. Since both methods aims to reduce the variance between runs, the combined results is worse in comparison to the preprocessing techniques separately.

5.4 Differences between classifiers

In line with our hypothesis, the results indicates a significant difference in execution time between the KNN classifier and the rest of the classifiers. However,

according to the results of section 4.3, the KNN classifier performed slightly worse compared to the both SVM classifiers. Hence, the long execution time of the KNN classifier is not compensated with better classification accuracy. Regarding the classification accuracy, the Naive Bayes and Decision tree classifiers performed worse in comparison to the other three classifiers. As mentioned in section 4.3.1, the results also shows an significant difference in sensitivity between the well and poor performing classifiers which also could be the reason for the difference between classification accuracy. In this research, sensitivity refers to the observations correctly classified as malign. In other words, a low sensitivity results in a large amount of patients incorrectly classified to have breast cancer of malignant type. This miss-classification can have both psychological and physical consequences for the patients concerned. Hence in regards to the sensitivity, there is apparent that both of the SVM classifiers and the KNN classifier performed best compared to the remaining classifiers.

5.5 Further research

Further research has to be conducted of finding the best possible classifier in classification of breast cancer. As the settings of classifier parameters are set automatically during runtime in this research, further investigation is required in optimizing classifiers by parameter tuning. Correctly optimized methods would improve the learning and consequently improve the classification performance [13]. Another example is the number of principal components used which possibly had a great impact on the results. As mentioned in section 5.1, a more exhaustive testing for different amounts of principal components is required to confidently draw conclusions regarding improvements of increased performance by PCA. Further analysis of PCA combined with the parameter tuning of classifiers could be beneficial in future research.

Aside from increased classification performance, quality assurance of the classification results is an aspect that could be taken into account. To ensure that the generated results are reliable, a second classifier, acting as a second opinion, could be introduced and combined to either confirm or deny the results. In further research, it would be interesting to compare instances between these classifiers and observe if the generated results are equivalent or not.

5.6 Effects of limitations

The limited amount of breast cancer datasets used for this research makes it difficult to interpret and draw correct general conclusions regarding the classification of breast cancer. Datasets may vary from the one used in this research in terms of number of features as well as the issue of handling missing data. This also brings to the topic of feature selection which was excluded in this research with respect to the scope. Another aspect is the limited amount of observations included and the used methods when collecting data. It is therefore difficult to tell whether the results gained can be generalized to populations in other countries and continents or if they are limited to the population observed.

As mentioned in 5.5, another limitation is the parameter tuning of internal settings for each classifier. The amount of different adjustments generates various optimized methods for different purposes which needs to be evaluated and taken into account of limitations. An example in this particular research is the automatically generated settings of Decision tree classifier which can be viewed in A. Note that the parameter *Max depth* is automatically set to *None* which generates a result of a fluctuating classification variance by default, as a consequence of over-fitting the data model to the training data. Consequently this gives the appearance that Bagging achieves significant results of the Decision tree classifier. However, limiting the depth of the decision tree classifier would have generated a classifier with lower variance. According to what we know about Bagging, this would probably have reduced the effect of using Bagging. This is just one example of the importance of the parameter setting and how it possibly could generate a misleading result.

5.7 Ethical aspects

Studies have proven that modern technology and intelligent computer programs can diagnose breast cancer more accurately than certified medical practitioners [5]. The process of diagnosing is also more effective and faster than regular physical examination. Along with the supplied benefits of AI technology, it also raises ethical issues that arises by integrating and utilizing computers in clinical practices and in medical education. Furthermore, addressing the balance of benefits and risks of integrating AI technology in medical healthcare. This raises important questions of the AI role in healthcare, which directly concerns the decision making in medical treatment, and how this will

impact the society. Consequently, questions such as: are the algorithms and computers accurate enough to make them more trustworthy than human medical practitioners in terms of diagnosing? How do one logically explain the results and output by using black-box algorithms? Furthermore, what are the legal issues in cases of medical malpractices with the utilization of complex algorithms? In the case of malpractice, who bears the responsibility? Another critical ethical and legal aspect which concerns patient integrity is the use and storage of delicate patient data to feed and train the algorithms. It is understandably that computers and AI technology comes with great benefits but also entails and raises important ethical and legal issues.

As computer-aided diagnostics consequently develops and integrates in healthcare, medical education has to adapt and prepare AI technology for medical students. Thus, this requires an extensive review of medical education concerning the preparation of future medical practitioners in the interaction and management of AI technology.

Chapter 6

Conclusions

A natural conclusion of the effects of utilizing the PCA and Bagging techniques is dependent on which classifier being used. In general, it is interesting that this research shows no considerable improvements regarding the classification accuracy and execution time using PCA or Bagging on the provided dataset. Particularly, the classifiers using both the preprocessing technique PCA and the ensemble method Bagging performed significantly worse. However, the only classifier benefiting and improved from Bagging was the Decision tree classifier. Yet, it did not achieve as good results as the other classifiers, in comparison, to be taken into account.

Regarding which machine learning method that is most suitable for analyzing and assessing whether a patient, diagnosed with breast cancer, has benign or malignant characteristics with respect to the classification performance, further research is required 5.5. This research shows that the two SVM classifiers performed slightly better than the remaining classifiers. As the tests were performed on only one dataset, with limited amount of observations, and the internal settings of classifiers were automatically set it is considered insufficient for a generalized result.

Bibliography

[1] Pedro Domingos. "A few useful things to know about machine learning". In: vol. 55, no. 10. 2012, pp. 1–4.

[2] Igor Kononenko. "Machine learning for medical diagnosis: History, state of the art and perspective". In: *Artificial intelligence in medicine* 23 (Sept. 2001), pp. 89–109. DOI: 10.1016/S0933-3657(01)00077-X.

[3] "Breast cancer". In: 2018. URL: https://www.who.int/cancer/prevention/diagnosis-screening/breast-cancer/en/.

[4] Shweta et al. Kharya. "Predictive Machine learning Techniques for Breast Cancer Detection". In: vol. 4(6). 2013, pp. 1023–1028.

[5] Chiara Longoni Carey K. Morewedge. "AI Can Outperform Doctors. So Why Don't Patients Trust It?" In: *Harvard Business Review* (). URL: https://hbr.org/2019/10/ai-can-outperform-doctors-so-why-dont-patients-trust-it.

[6] Gareth James et al. *An Introduction to Statistical Learning: with Applications in R*. Springer, 2013. URL: https://faculty.marshall.usc.edu/gareth-james/ISL/.

[7] Ahmad B A Hassanat. "Two-point-based binary search trees for accelerating big data classification using KNN". In: *PubMed* 13 (). DOI: 10.1371/journal.pone.0207772.

[8] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[9] "Basic Information About Breast Cancer". In: 2019. URL: https://www.cdc.gov/cancer/breast/basic_info/index.htm.

[10] Niklas. Price Thony. Lindqvist. "Evaluation of Feature Selection Methods for Machine Learning Classification of Breast Cancer". In: 2018.

[11] Bhagirathi Halalli and Aziz Makandar. “Computer Aided Diagnosis - Medical Image Analysis Techniques”. In: Jan. 2018. ISBN: 978-953-51-3732-0. DOI: 10.5772/intechopen.69792.

[12] Hao Karen. “What is machine learning?” In: 2018. URL: https://www.technologyreview.com/s/612437/what-is-machine-learning-we-drew-you-another-flowchart/.

[13] Zakia Salod and Yashik Singh. “Comparison of the performance of machine learning algorithms in breast cancer screening and detection: A protocol”. In: *Journal of Public Health Research* 8 (Dec. 2019). DOI: 10.4081/jphr.2019.1677.

[14] Diana Dumitru. “Prediction of recurrent events in breast cancer using the Naive Bayesian classifcation”. In: *Analele Universităţii din Craiova. Seria Matematică Informatică* 36 (Sept. 2009).

[15] Mehmet Akay. “Akay, M.F.: Support vector machines combined with feature selection for breast cancer diagnosis. Expert systems with applications 36(2), 3240-3247”. In: *Expert Syst. Appl.* 36 (Mar. 2009), pp. 3240–3247. DOI: 10.1016/j.eswa.2008.01.009.

[16] Graziella Orrù et al. “Machine Learning in Psychometrics and Psychological Research”. In: *Frontiers in Psychology* 10 (2019).

[17] Mejia Niccolo. “Artificial Intelligence in Medical Robotics – Current Applications and Possibilities”. In: 2019. URL: https://emerj.com/ai-sector-overviews/artificial-intelligence-medical-robotics/.

[18] Ahmed Osmanović et al. “Machine Learning Techniques for Classification of Breast Cancer”. In: May 2019, pp. 197–200. ISBN: 978-981-10-9034-9. DOI: 10.1007/978-981-10-9035-6_35.

[19] J. D. Hunter. “Matplotlib: A 2D graphics environment”. In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.

[20] Michael Bironneau and Toby Coleman. “Machine Learning with Go Quick Start Guide: Hands-on techniques for building supervised and unsupervised machine learning workflows.” In: Pubt publishing Ltd., 2019, p. 45.

[21] Alaa Tharwat. “Classification assessment methods: a detailed tutorial”. In: (Sept. 2018).

Appendix A

Classifier parameters

Classifier	Parameter	Value
Naive Bayes	Type	Gaussian
	Priors	None
Support Vector Machines	Penalty	l2
	Loss	Squared hinge
	Dual	True
	Tolerance	1e-4
	Multi class	Ovr
	Fit intercept	True
	Verbose	0
	Random state	None
	Max iterations	1000
	Intercept scaling	1
SVM Linear	C	0.025
SVM Radial	C	1.0
K-Nearest Neighbor	Number of neighbors	5
	Weights	Uniform
	Leaf size	30
	Metric	Minkowski (Euclidean distance)
	Metric params	None
	Power parameter	2
Decision Tree	Criterion	Gini
	Splitter	Best
	Max depth	None
	Min samples split	2
	Min samples leaf	1
	Min weight fraction leaf	0
	Max features	None
	Random state	None
	Max leaf nodes	None
	Min impurity decrease	0
	Class weight	None
	CCP alpha	0
Bootstrap aggregating	Base estimator	None
	Number of estimators	10
	Max samples	1.0
	Max features	0.7
	Bootstrap	True
	Bootstrap features	False
	Warm start	False
	Random state	None
	Verbose	0

Table A.1: Parameters used for each classifier

TRITA -EECS-EX-2020:404