UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

ANALYSIS OF CLINICAL CANCER DATA USING MACHINE LEARNING ALGORITHMS

Diploma Thesis

IOANNA TSOTRA

Supervisor: Michael Vassilakopoulos

Volos 2020

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

ANALYSIS OF CLINICAL CANCER DATA USING MACHINE LEARNING ALGORITHMS

Diploma Thesis

IOANNA TSOTRA

Supervisor: Michael Vassilakopoulos

Volos 2020

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΑΝΑΛΥΣΗ ΚΛΙΝΙΚΩΝ ΔΕΔΟΜΕΝΩΝ ΤΟΥ ΚΑΡΚΙΝΟΥ ΜΕ ΤΗ ΧΡΗΣΗ ΑΛΓΟΡΙΘΜΩΝ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

Διπλωματική Εργασία

ΙΩΑΝΝΑ ΤΣΟΤΡΑ

Επιβλέπων: Μιχαήλ Βασιλακόπουλος

Βόλος 2020

Approved by the Examination Committee:

Supervisor **Michael Vassilakopoulos**
Associate Professor, Department of Electrical and Computer Engineering, University of Thessaly

Member **Eleni Tousidou**
Laboratory Teaching Staff, Department of Electrical and Computer Engineering, University of Thessaly

Member **Panagiota Tsompanopoulou**
Associate Professor, Department of Electrical and Computer Engineering, University of Thessaly

Date of approval: 20-9-2020

Acknowledgements

Reaching the final destination of this endeavor, I would like to thank some people who contribute to accomplishing my academic goals.

First of all, I would like to thank Mrs. Eleni Tousidou for the valuable help and guidance she has given me all this time to complete this thesis. Also, I would like to express my sincere acknowledgements to Professor Michael Vassilakopoulos for undertaking the supervision of my thesis and for the opportunity he gave me to do so.

Moreover, I would like to express my heartfelt gratitude to my amazing friends, both the ones I did in Volos but also the ones I had since my school years, for standing by my side all this time. I am really grateful to all of them.

Last but not least, I would like to thank my parents and my brother for their unconditional love and support all these years that made me the person I am today. They encourage and help me with their own unique way to achieve my goals, and they will always be the most important thing of my life.

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

«Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism».

The declarant

IOANNA TSOTRA
15-9-2020

Abstract

Cancer as a disease of our time plagues millions around the world, as there are a multitude of incidents and deaths every day, including a wide range of ages. More specifically, according to recent statistical studies, breast cancer has proven to be one of the most deadly forms for the female population. Although important steps have been taken in early detection and diagnosis to investigate a more effective treatment for cancer, its prognosis remains the most important step in its treatment. As prognosis, is defined the likelihood of cancer, relapse or even survival. In recent years, with the rapid development of technology and the techniques of Machine Learning, several prediction models have been developed with relatively high accuracy. In this thesis, the aim is to predict the type of breast cancer (benign or malignant). Thus, for the construction of the models, six Machine Learning algorithms were tested, applied in two data sets. The algorithms used were: Logistic Regression, K-Nearest Neighbors, Naïve Bayes, Support Vector Machine, Random Forest and Neural Networks. The results showed that the Support Vector Machine and Neural Networks algorithms worked more efficiently for these data.

Keywords

Breast Cancer Prediction, Machine Learning, Data Analysis, Neural Netwroks, Python

Περίληψη

Ο καρκίνος ως ασθένεια της εποχής μας μαστίζει εκατομμύρια ανα τον πλανήτη, καθώς σημειώνονται καθημερινά πληθώρα περιστατικών αλλα και θανάτων, περιλαμβάνοντας ένα μεγάλο εύρος ηλικιών. Πιο συγκεκριμένα, σύμφωνα με πρόσφατες στατιστικές μελέτες, ο καρκίνος του μαστού αποδείχτηκε μια απο τις πιο θανατηφόρες μορφές για τον γυναικείο πληθυσμό. Παρότι έχουν γίνει σημαντικά βήματα για την έγκαιρη ανίχνευση και διάγνωση με στόχο την έρευση μιας αποτελεσματικότερης θεραπείας ενάντια στον καρκίνο, η πρόγνωσή του παραμένει ακόμα το σημαντικότερο στάδιο για την αντιμετώπισή του. Ως πρόγνωση μπορούμε να ορίσουμε την πρόβλεψη πιθανότητας εμφάνισης καρκίνου, υποτροπιασμού ή ακόμα και επιβίωσης. Με την πρόοδο της τεχνολογίας τα τελευταία χρόνια και των τεχνικών Μηχανικής Μάθησης, έχουν αναπτυχθεί αρκετά μοντέλα πρόβλεψης με σχετικά μεγάλη ακρίβεια. Στην παρούσα διπλωματική, ο στόχος είναι η πρόβλεψη του τύπου καρκινώματος του μαστού (καλοήθης ή κακοήθης). Έτσι, για την κατασκευή των μοντέλων δοκιμάστηκαν έξι αλγόριθμοι Μηχανικής Μάθησης, εφαρμοσμένοι σε δυο σύνολα δεδομένων. Οι αλγόριθμοι που χρησιμοποιήθηκαν ήταν οι εξής: Logistic Regression, K-Nearest Neighbors, Naïve Bayes, Support Vector Machine, Random Forest και Neural Networks. Τα αποτελέσματα που προέκυψαν έδειξαν πως οι αλγόριθμοι Support Vector Machine και Neural Networks λειτούργησαν πιο αποτελεσματικά για τα συγκεκριμένα δεδομένα.

Λέξεις Κλειδιά

Πρόβλεψη καρκίνου, Μηχανική Μάθηση, Ανάλυση Δεδομένων, Νευρωνικά Δίκτυα, Python

Table of contents

List of figures

List of tables

Abbreviations

ANN	Artificial Neural Networks
AUC	Area Under the Curve
KNN	K-Nearest-Neighbor
ML	Machine Learning
MLP	Multilayer Perceptrons
MSE	Mean Squared Error
NN	Neural Networks
PCA	Principal Component Analysis
SVM	Support Vector Machine
SVR	Support Vector Machine algorithm for Regression
WBC	Breast Cancer Wisconsin (Original)
WDBC	Wisconsin Diagnostic Breast Cancer (WDBC)

Chapter 1

Introduction

1.1 Description of the problem

Cancer nowadays is one of the most common causes of death all around the world. As a disease, it causes the cells to mutate in a process that can take up to 10 years. The main factors that can cause cancer are smoking, diet and age. The most dangerous form of cancer-related mortality is lung cancer regardless of the patient's sex, prostate cancer in men and breast cancer in women.

More specifically, according to NIH (National Institute of Health) in the USA the number of estimated new cases of breast cancer in 2020 so far is approximately 276,480 and the deaths are estimated around 42,170. In the last decades the relative survival rate is about 90%. [1]

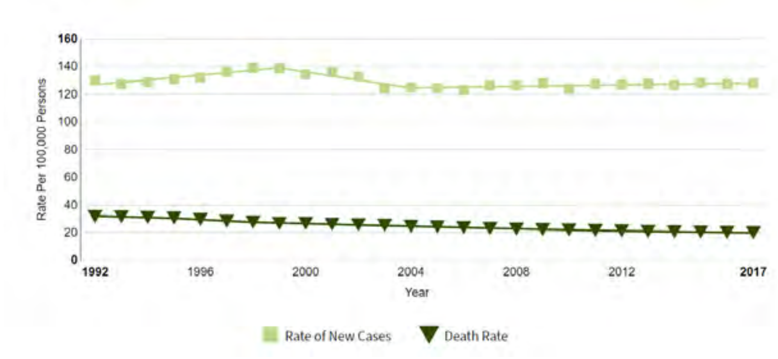


Figure 1.1: New cases and Death Rate in US from SEER (Surveillance, Epidemiology and End Results Program) [1]

As we can see, breast cancer is the most common cancer among women worldwide. Some of the most important factors are the following:

- Family history: A woman whose close relative had breast cancer especially at a young age (<30), has a higher risk of developing cancer
- Age
- Personal history of breast cancer
- Genetic Factors: Certain genetic mutations, including changes to the BRCA1 and BRCA2 genes are an important factor for developing breast cancer
- Childbearing and menstrual history: If a woman is old when she has her first child, then the risk of breast cancer is high

The summary stage system of breast cancer used for descriptive and statistical analysis of tumor registry data, according to the SEER (Surveillance, Epidemiology, and End Results Program) is the following [11]:

- In situ stage mentions to the attendance of abnormal cells that are confined to the layer of cells where they created.
- Local stage refers to advanced cancer that is constricted to the breast.
- Regional stage mentions to cancer that has feast to surrounding tissue and/or close lymph nodes.
- Distant stage refers to cancer that has spread to faraway organs and/or lymph nodes, including nodes above the collarbone.

Traditionally, breast cancer is typically identified either throughout screening, before symptoms have matured, or after a woman notices a lump. Most masses seen on a mammogram turn out to be benign. In case a tumor is suspected to be malignant, a tissue for microscopic analysis is normally acquired from biopsy [11].

Regarding breast cancer, while incidence rates are increasing throughout the world, mortality rates have been decreasing or have remained relatively stable over the last decades. Both early diagnosis of mammography and progress in treatment have contributed to this.

Simultaneously with the cumulative progress of biomedical and computer technologies, varied clinical factors related to breast cancer have been noted. To intercept the considerable increase of breast cancer, many researchers have considered using patient clinical records to predict the type of patient's tumor, the survival probability or even the likelihood of malignant tumor metastasis [12]. Extracting knowledge from oncology data is a difficult but at the same time interesting part of research due to the enormous amount of data and the many features that must be pre-processed and analyzed during the research. In Data Science field, Machine learning and Data mining algorithms are widely used in prognosis and analysis to make decisions. These algorithms allow precision and fast classification of breast cancer based on images e.g. for a surgical biopsy or on numerical data, such in this thesis. Thus, accurate classification of benign or malignant tumor is a very important subject and can help to prevent the development of breast cancer, promote timely clinical treatment to patients or even prevent undergoing unnecessary treatment.

1.2 The Aim

The aim of this thesis is to predict breast cancer using and combining data mining and machine learning algorithms. Based on two real-life datasets, with patient's information, we discover whether the tumor is malignant or benign. Simultaneously, we compare our results with other researches, trying to produce an effective and strong prediction model. This thesis will help in the medicine or biomedical field by giving the chance to have an early prognosis about the kind of patient's tumor, so can improve the chance of survival or even more prevent patients undergoing unnecessary treatments. Furthermore, it may contribute to other machine learning researches and finally build a very strong predicted model.

1.3 Chapters' Organization

In Chapter 1, there is the introduction of the thesis, where the problem is analyzed and the purpose of this thesis is described.

In Chapter 2, the related work is described, while in Chapter 3 all the theoretical background of machine learning and machine learning algorithms used in this research are included, as well as the theory by which the performance of these algorithms is measured.

Chapter 4 provides information about the two data sets and the software tools used.

In Chapter 5, the methodology and the details about the implementation of this thesis are presented.

In Chapter 6 the results of the models as well as some comparisons made are reported. Finally, in Chapter 7 the conclusions and the future extensions of the thesis are presented.

Chapter 2

Related Work

This chapter will include information on some of the related work and research that has been done and has helped with this particular thesis.

2.1 Related Work

As mentioned above, the valid diagnosis and treatment of breast cancer is a very important subject, especially for women. With the development of knowledge discovery methods, there are varied studies that implemented data mining and machine learning techniques to diagnose breast cancer cases based on patient information.

First of all, Saleema et al's research concerned the application of some classification algorithms on the SEER data set by using MATLAB. In this data set, periodical incidences of breast, respiratory and mixed cancer have been recorded during the year 2000-2007. In this study, the importance of the sampling method was emphasized during the application of data mining algorithms. The algorithms examined were Naïve Bayes, K-Nearest neighbor and Neural Networks and the results showed that regardless of the algorithm used, its accuracy tends to increase relatively to the sample size.

Wang and Yoon's (2015) research was about applying a series of algorithms to ready-made data sets (dataset: Wisconsin Breast Cancer Database-WDC (1991) and Wisconsin Diagnostic Breast Cancer-WDBC (1995)) and measuring the accuracy of the results. Here the most common statistical tool for dimension reduction, PCA (Principal Component Analysis) has been used, which can transform possible correlated variables into uncorrelated variables. In the research four data mining methods as well as eight hybrid models with the combination of PCA method are tested. The greatest accuracy was shown in the application of the Neural Networks and SVM (Support Vector Machine) algorithms with PCA, but the examination of the statistical significance showed that these are not statistically significant.

One more significant study was that of Asri et al's in 2016. Here with the data mining tool, WEKA, they examined the results of SVM (Support Vector Machine), Decision Tree (C4.5), Naïve Bayes and KNN (k Nearest Neighbors) algorithms on Wisconsin Breast Cancer Dataset (1991). They try to compare the efficiency and effectiveness of those four algorithms in terms of accuracy, precision, sensitivity and specificity to find the best classification accuracy. The results show that SVM method gives the highest accuracy with lowest error rate.

Punjani et al's (2017) study is focused on improving the practical application of real-time data mining algorithms and did not provide numerical results on the accuracy of the methods used. After all the theoretical analysis, Decision Tree method is used to to classify 50 females patients' data with the aim to examine the risk of cervical cancer appearance. The results of the study described the high accuracy of the classification methods, confirming their usefulness in predicting the occurrence of cancer in real conditions.

Sahu et al's (2018) study was also an interesting approach. Their goal was to help in the accurate classification of benign tumors that can prevent patients from unnecessary treatments. In this paper they have purposed a prediction model by combining Artificial Intelligent based learning technique with a multivariate statistical method. To predict breast cancer, they suggest a hybrid method with the combination of PCA technique and Artificial Neural Networks (ANN) by using R and MATLAB for simulation purpose. The algorithm was tested in Wisconsin Breast Cancer Dataset from UCI. Sensitivity and F measure have achieved the better evaluation results. Their conclusion was that Machine Learning and more specifically ANN plays a major role in the detection of cancer diagnosis.

Last but not least, is the study of S.Vanitha and Dr.P.Balamurugan in 2017. The aim of this paper was to analyze medical data and find out the better classification algorithm. For the implementation they applied SVM and NN (Neural Networks) in 2 datasets. The first one was Cleveland heart disease data set with the classification target value be "disease" and "non-disease", and the second one was Wisconsin Original Breast Cancer data set with target values "benign" or "malignant". More specifically, for NN classifier and breast cancer dataset, ten features are used as training parameters for the neural network, using the scaled conjugate gradient algorithm with 20 hidden neurons for 1000 epochs. In SVM algorithm an optimum linear separating hyperplane to separate two set of data is used. The outcome of this study demonstrates that Neural Networks effectively undertake these kinds of medical data with high accuracy.

Chapter 3

Theoretical Background

3.1 Introduction

In this chapter, information on what Machine Learning deals with is provided, along with some basic concepts of the theoretical background that lie behind the machine learning algorithms. We will also refer to the various techniques available for data processing and finally we will report on the ways by which we measure the performance of machine learning algorithms.

3.2 Machine Learning

Machine Learning is a field of Computer Science and is defined as the process by which a computer system, through the continuous execution of a command, improves its performance without having to re-program. In 1997 Tom Mitchell in his book Machine Learning defined: “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.” [13]

In the field of Machine Learning, three basic categories of learning have been developed that can been correlated with the ways a person learns. These categories are Supervised Learning, Unsupervised Learning and Reinforcement Learning:

• Supervised Learning: It describes the process by which the algorithm is constructed with input data that can also contain the actual result expected by that algorithm. This way, the algorithm can have the right results as a ”educator”, and we can measure its performance.

Supervised Learning can be used with classification, interpretation and prediction problems.

• Unsupervised Learning: It describes the process when the algorithm constructs a model without knowing the desired output. It is used in association analysis and clustering problems.

• Reinforcement Learning: The algorithm learns a strategy of action through direct interaction with the environment. This type of learning is used mainly in planning problems like in Robotics [13] .

Another categorization that is related with the result we seek using machine learning algorithms, is the following:

• Classification methods: Their aim is to sort the data into two or more categories. This method belongs to supervised learning. Usually, data that are included in these methods are categorical data.

• Clustering methods: They try to classify the data, like classification methods, but without the knowledge of the classes.

• Regression methods: They are used to predict a specific number among continuous values. This category belongs also to supervised learning.

• Dimensionality reduction: This method is used to simplify data by converting space into smaller dimensions.

Considering all the above, the problem studied in this thesis belongs to Supervised Learning and consists a Classification problem, where we try to efficiently predict the class a breast tumor belongs to, malignant or benign.

3.3 Split into Train and Test sets

For a classification problem, test error is the most important measure to evaluate classification performance. However, when a dataset is given, all the data are expected to be used to train the model, which can enhance the model to obtain a more effective and robust reliable learning result [12]. Several theories have been developed as to what is the most appropriate combination between training and test data sets. The most common use is to split the data set into 75% data for algorithm training and the remaining 25% for control [14]. Still, we will often see a smaller percentage being used to train the algorithm with the main objective of avoiding overfitting. Thus the algorithm uses the training data to construct the model and then the test data through which it results and how efficient the algorithm is.

3.4 K-folds Cross Validation

Once the data has been divided into two subsets one for training and one for testing then the algorithm should calculate the parameters of the model. For this reason it will use the training data. But the way the data will be split, being random, will probably not give us the best possible result for the prediction, and the reason is that the training data may not be appropriate. To solve this problem we will use the Cross Validation method and specifically the K-fold Cross Validation. With this method we construct according to the value we have given to the parameter k, so many sets of data, from training and test datasets separated randomly [14].

In the following figure for example, if we set the variable k=5, 5 sets of data will be created, each with a random but different way each time in training and control data.

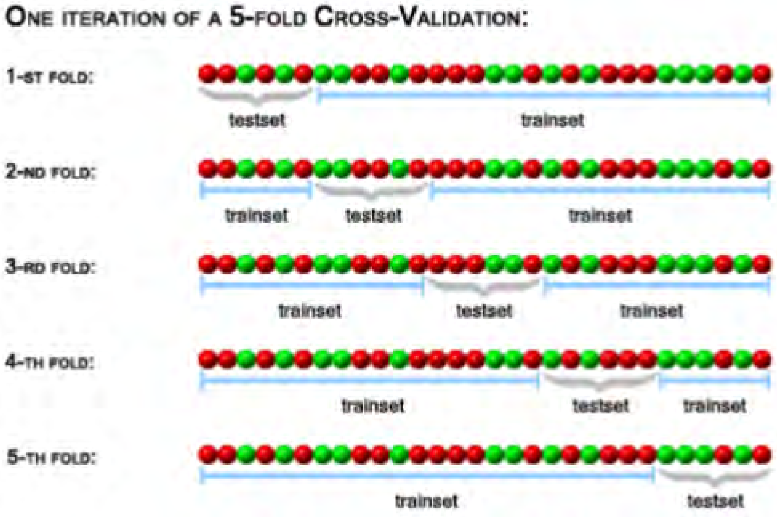


Figure 3.1: K-folds Cross Validation [2]

3.5 Overfitting and Underfitting

A fairly common phenomenon which is profound in machine learning models is that of overfitting. By this term we designate a model when the model has “learned” the training data very well so that it cannot make good predictions about the test data. So, this model has come to learn everything from training data even the noise. The problem is that these concepts do not apply to new data and negatively impact the model’s ability to generalize. Overfitting is more likely with non-parametric and nonlinear models that have more flexibility when learning a target function. In the opposite case, when an algorithm does not ”learn” the training data well, we have the phenomenon of underfitting which can usually be tackled by the introduction of more training data. There has been a lot of discussion about whether the overfitting phenomenon is good, and the views are divergent [15].

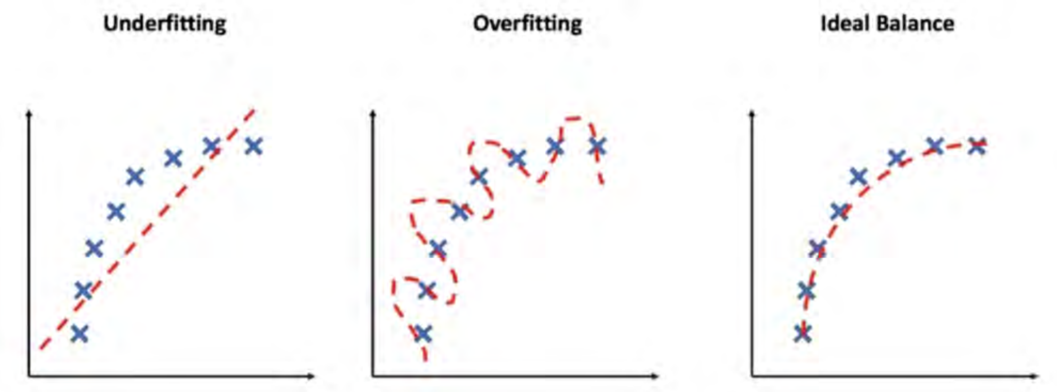


Figure 3.2: Overfitting and Underfitting Problem [3]

3.6 Machine Learning Algorithms

In this subsection all the theoretical background of Learning Algorithms are used in this thesis is presented.

3.6.1 Logistic Regression

Logistic Regression is a statistical model which was used in early in this century in biomedical science and generally in binary classification problems, with the goal to model the probability of a random variable being 0 or 1 given experimental data.

More specifically, in Machine Learning, logistic regression model incomes real-valued inputs, and makes a prediction regarding the probability of the input belonging to the default class. To identify in which class a data belongs to, a threshold can be set. Based in this threshold, the obtained estimated probability is classified into classes. For example, if our predicted value is equal or greater to 0.5 (threshold), the output is the prediction for the default class, else the prediction indicates the other class [16] [17]. The decision boundary can be linear or non-linear. The output of the predictions is transformed using the logistic function:

$$Output = b0 + b1 * X1 + ... + bn * Xn \tag{3.1}$$

$$P(class = 1) = \frac{1}{(1 + e^{-output})} \tag{3.2}$$

The coefficients ('b') of the logistic regression algorithm must be estimated from the training data. One effective way to estimate these values is to use Gradient Descent method. The goal of training a machine learning model is to minimize the error between the real values and the predictions by fiddling with the trainable parameters. So, in this method we repeatedly calculate the error for each prediction until the error converges in a small number of iterations [16] [17, 18].

3.6.2 K- Nearest Neighbors

K-Nearest Neighbors (KNN) is a simple supervised Machine Learning algorithm that can solve both classification and regression problems. The algorithm uses a distance function to classify a new case. This new case will be allocated to the most common class among its K nearest neighbors measured by a distance function [4, 5]. The three distance measures that are used to determine which of the K neighbors are most similar to a new input are the following:

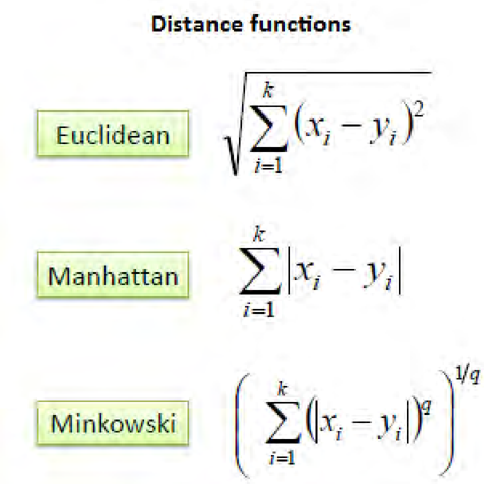

Distance functions
Euclidean
$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$
Manhattan
$\sum_{i=1}^{k}|x_i - y_i|$
Minkowski
$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$

Figure 3.3: Distance Functions [4]

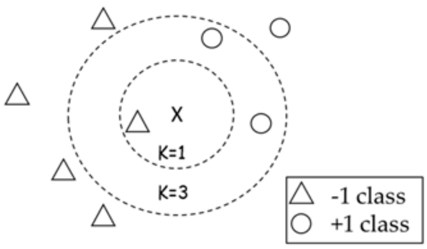


Figure 3.4: KNN Algorithm [5]

3.6.3 Naïve Bayesian

Naïve Bayesian classifier is based on the application of Bayes theorem, which is the following:

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \tag{3.3}$$

Where:

- P(A) is the prior probability of class
- P(A|B) is the posterior probability of class (target) given predictor (attribute)
- P(B|A) is the likelihood which is the probability of predictor given class
- P(B) is the prior probability of predictor

After calculating the posterior probability for several different classes, selects the class with the highest score.

3.6.4 Support Vector Machine (SVM)

One of the most popular and widely discussed methods are Vector Support Machines or SVM, which was invented in 1963 by Vladimir N. Vapnik and Alexey Ya. Chervonenkis and became more popular around 1990 [19].

Classification

Initially support vector machines have begun to be used to divide the data into two categories or classes. Their main goal is to create a hyperplane, which splits the two classes, trying maximize the margin between the two classes.

As we can see in the left diagram of the Figure 3.5 there are several ways to split the data into two classes or categories. The effectiveness of SVM algorithm lies in its ability to find

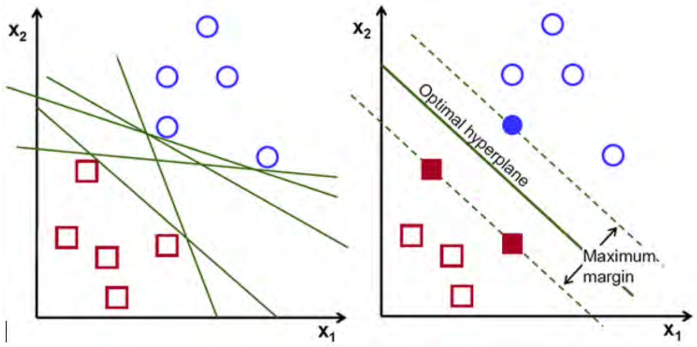


Figure 3.5: Graphical representation of the line of the SVM algorithm [6]

that line which will best divide our data. Consequently, we have two lines defining the space and are called boundary lines, the line found in the space created by these two lines called optimal hyperplane and finally the support vectors, that help us construct the final model. The equation of the line dividing the space into two classes is: $wx + b = 0$. Moreover, we will use two support vectors: $wx + b = -1$ and the $wx + b = +1$, and we will have to find the distance between them by taking the vertical distance as follows:

$$d = \frac{2}{|| \, w \, ||}$$

As we mentioned before our goal is to maximize the distance d and to achieve this, we can just minimize the previous equation. So:

$$min \frac{(|| \, w \, ||)^2}{2}, y_i(wx_i + b) \geq 1, \forall x_i$$

Because the above formula could be applied in an ideal sorting problem and not easily in real-world data it is truly necessary to “relax” the constraint of maximizing the margin of the line that separates the classes. In this way, the desired hyperplane is a softer surface ξ_i making as fewer misclassification mistakes as possible. Also, there is a tuning parameter C that defines the amount of violation of the margin allowed [20].

$$min(\frac{(|| \, w \, ||)^2}{2} + C \sum_{i=1}^{N} \xi_i), y_i(wx_i + b) \geq 1 - \xi_i, \forall x_i, \xi_i \geq 0$$

Regression

The purpose of regression in a two-dimensional (2D) problem is to create a straight line with the aim of minimizing the errors between the real values and the line's ones. This problem can be solved with linear regression method. However, in the case where the data are not linear, SVM algorithm for regression (SVR) is an effective solution.

The main idea is remained the same: to maximize the margin between the observations and the hyperplane. Firstly, in the case of SVR algorithm, the support vectors have the role of margin of tolerance (epsilon). The model's formula is quite similar to the classification's:

$$min\frac{(\parallel w \parallel)^2}{2}, y_i - wx_i - b \leq \epsilon, wx_i + b - y_i \leq \epsilon$$

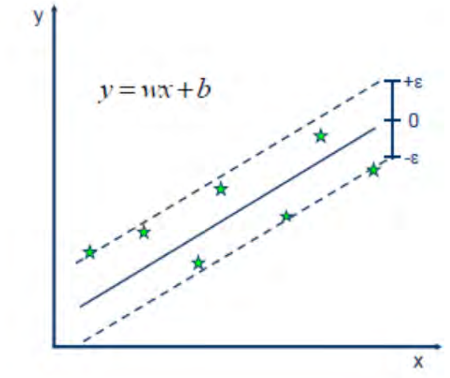


Figure 3.6: Graphical representation of the line of SVR algorithm [7]

Like above we have a pair of variables ξ_i and ξ_i^*, which show us how far the observations are from the support vectors. As we can see from the figure, if the observations are within the space defined by the support vectors then these two variables are zero [7]. So, the new model will be:

$$min(\frac{(\parallel w \parallel)^2}{2} + C\sum_{i=1}^{N}\xi_i + \xi_i^*),$$

$$y_i - wx_i - b \leq \epsilon + \xi_i, \forall x_i, wx_i + b - y_i \leq \epsilon + \xi_i^*, \forall x_i, \xi_i, \xi_i^* \geq 0$$

Kernel

The advantage of SVR algorithm over the Linear Regression is that it can be applied to non-linear models. Generally, the implementation of SVM algorithm for linear and non-linear problems is achieved by kernel functions, which transform the problem data into a space larger than they already exist [21].

The most common kernel functions are the following:

- Linear:

$$K(x_i, y_i) = x_i y_i + \gamma$$

- Polynomial:

$$K(x_i, y_i) = (x_i y_i + 1)^p$$

- Radial:

$$K(x_i, y_i) = e^{-\gamma(x_i - x_j)^2}$$

- Gaussian:

$$K(x_i, y_i) = e^{\frac{-1}{2\sigma^2}(x_i - y_i)^2}$$

3.6.5 Random Forest

Random Forest is a supervised, ensemble learning algorithm, which can be used in both classification and regression problems. As its name suggests, the algorithm builds a forest with a number of trees. The higher the number of trees in the forest, the higher the accuracy results. Each one tree in the Random Forest declares a class prediction and the class with the most votes becomes our model’s prediction. The key is the low correlation between the trees-models, because each tree protects all the other trees from its individual error. So, the “wrong” trees as a group may be able to move in the correct direction, like the “right” trees. It uses random features when building each one tree to try to create an uncorrelated forest of trees. Random Forest its is also a good idea for the overfitting problem [20] [22] [23].

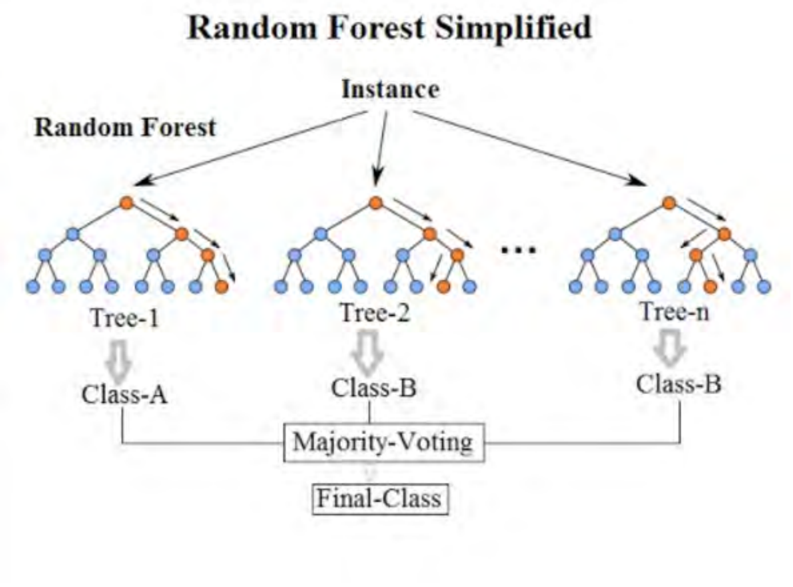


Figure 3.7: Random-Forest Algorithm [8]

3.7 Principal Component Analysis

Principal Component Analysis or PCA is a dimensionality reduction technique to signify the data into a lower dimensional space, but still characterizes the intrinsic relationships in the data. This statistical method will have a growing impact on modern neuroscience and biology field, because of its usefulness. More specifically, PCA tries to find “components” that capture the maximal variance within the data, like in the following image:

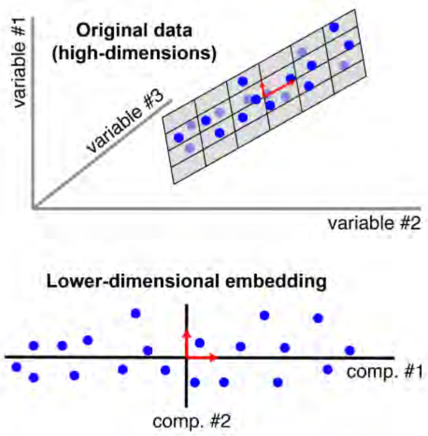


Figure 3.8: Its blue point corresponds to an observation. Here PCA reduces the dimensionality for 3 to 2

So the huge dataset can be reduced and can be expressed within few numbers of variables. The main function of PCA is to detect the patterns in dataset and find similarity and differences between each individual attribute. The algorithm uses the concepts of variance matrix, covariance matrix, eigenvalues and eigenvectors pairs to perform PCA, as long as a set of eigenvectors and its respectively eigenvalues as a result.

The first principal component is the feature that produced the highest variance.The feature that is accountable for the second highest variance is regarded the second principal component, and so on. The principal components, is very predominant, not to have any correlation between them [24] [25].

3.8 Artificial Neural Networks

Artificial Neural Networks or ANN is an information processing paradigm based on the function of human brain. Their basic function is very similar to that of human neurons in a simpler form, of course. They are made up of elements that are interconnected, such as the neurons in the human brain, receive stimuli from the entrance and according to these stimuli learn and react [26].

The technology of ANN is a relatively new area in the field of machine learning as their study and research has begun in the last 40 years. However, their study is well advanced, and today they are one of the best solutions for classification and regression problems, especially for building predictive models.

As in the human brain, also in neural networks there is an interconnected network of computational nodes (neurons). An example of a node is shown in the figure below:

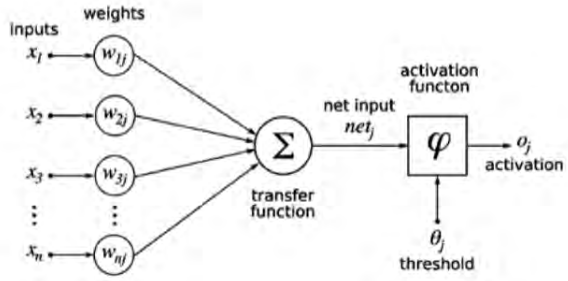


Figure 3.9: Neural Network Structure

In ANN there are three types of neurons [27]:

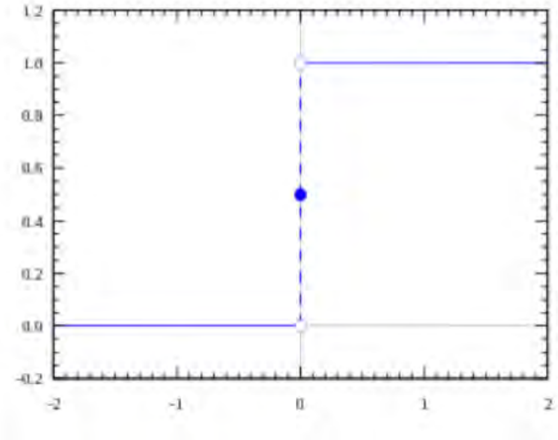
- Input layer: The intent of input neurons (visible layer) is to import into the network the original data or results of a previous layer. Usually the number of input neurons is equal to the attributes of the data set we want to import. In this neuron, no other process is performed except to forward the input data to the hidden neurons.

- Hidden layer: They are called hidden layers because they are not directly exposed to the input. Here the data are imported from the input neurons and some transformations are applied to them. Initially, each one multiplies by a certain weight, and then they are all added together, and a number is obtained. The number of hidden neurons varies and there are several theories about how this number can be selected.

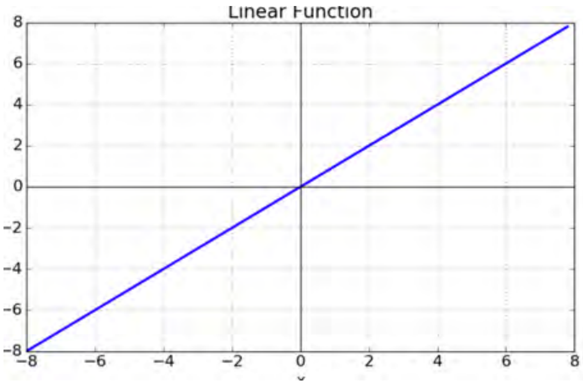
- Output layer: Data from the hidden neurons are first passed through the activation function, which will be activated when the sum of data exceeds the threshold we have set. Then the output of the neural network will occur. The output will be a value or a vector of values.

The data before leaving the output of the Neural Network first pass through an Activation function. Their main purpose is to convert an input signal of a node in an ANN to an output signal [26] [27]. The most common activation functions are the following [28] :

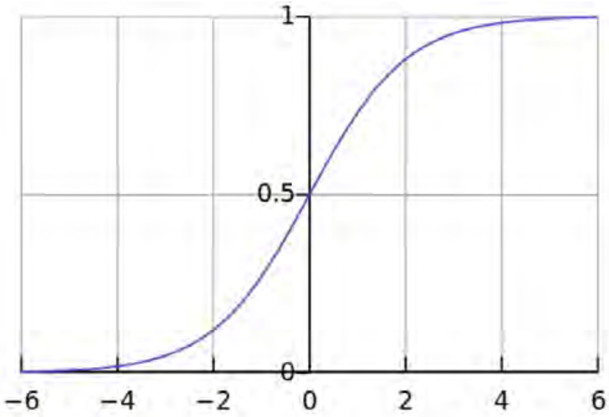
- Step Function or Threshold Function: If the output of the function is greater than the limit (bias) then the output of the activation function is 1, otherwise it is 0.



- Linear Function: The linear function is mainly used to solve linear regression problems and, as most of the functions, enters before the output neuron ($A = c * x$)



- Sigmoid or Logistics Function: It is a non-linear function that effectively solves non-linear problems. It is one of the most well-known functions used in multilevel neural networks.



3.8.1 Multilayer Perceptrons- MLP

The most well-known and useful type of neural networks is the Multilayer Perceptron or MLP for short. An MLP consists of two or more perceptrons. In the Figure below an MLP network is presented, with an input layer, two hidden layers and an output layer. Here, each one neuron of a network is interconnected with all the neurons from the previous layer [27, 9].

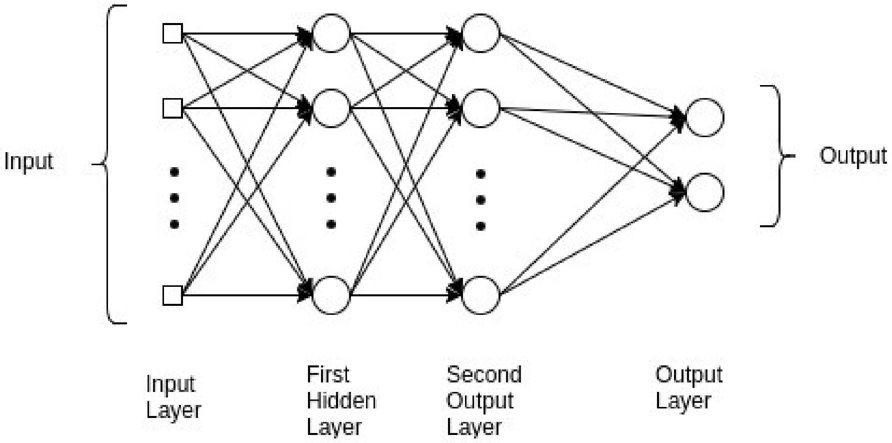


Figure 3.10: Multilayer Network [9]

Another characteristic of MLP networks is the use of each neuron of the hidden level as an activation function. The most usual is the application of the sigmoid function (or logistics function). The reason we prefer to use the sigmoid function is that it is a continuous function and therefore differentiable in contrast to the step function that has binary values. And the reason we want the function to be differentiable is that all the optimization methods we will use involve derivatives.

The training of these MLP networks is done through the process of adjusting the weights until one of the given optimization criteria is fulfilled. Our goal is to make the model learn from the data we have given it for training, so that it can predict the values of the next data that we will give it. The algorithm that uses such neural networks is the Error-Back Propagation algorithm. A key feature of this algorithm is that it is trained with supervision, more specifically, there is a goal that we want the algorithm to achieve [29].

The training process begins with the random initialization of the weights of each neuron. Then the process proceeds with two passages of the network, one forward and one backward. In the forward process each element of the input passes through the hidden levels and then ends up in the output of the network giving a value to the dependent variable. In this passage

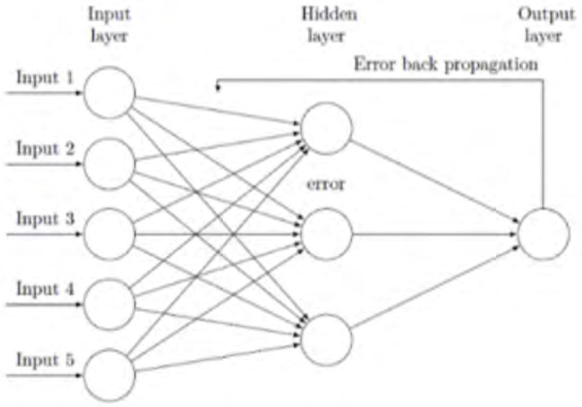


Figure 3.11: Neural Network with Error-Back Propagation

the weights remain constant. After passing all the input elements and giving a result passing through the output level then we can calculate the model error by subtracting the values we have found from the real values using a cost function. Then, to calculate how much each weight affects the error we need to calculate the partial derivatives of the final node and then apply the gradient descent algorithm. The form of the function used to renew the weights is as follows:

$$w = w - \eta \bigtriangledown_w J(w)$$

Then this information is transferred to the immediately previous level where we must calculate the partial derivatives by using the previous information about the weights in order to determine their value. This procedure continues until we reach the input level, which will obviously not change.

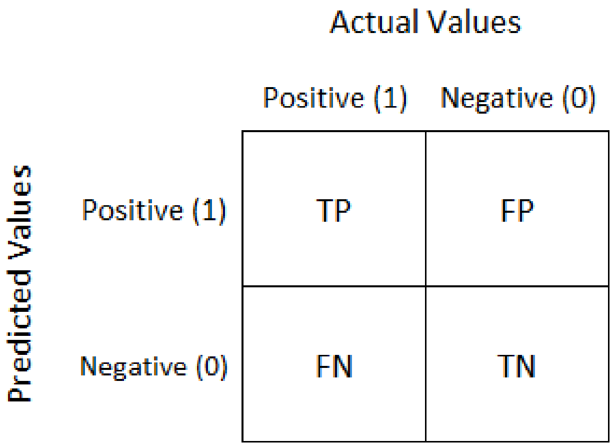
The above process will stop as soon as the number of iterations set by the user is completed, or when the network error is less than the limit set by the user again.

3.9 Algorithm Evaluation metrics

One way to evaluate whether a model is good is to measure its performance against its results. It is achieved by calculating the error, which is nothing but the difference between the actual and the forecast value given by the model. The above method applies to supervised learning models and normally the smaller the error, the better the performance of the model.

Classification problems are maybe the most frequent and familiar type of machine learning problem and as such there are innumerable metrics that can be used to evaluate predictions for these problems. Some of them are the above:

- Area Under ROC Curve: It is a performance metric for binary classification problems. Particularly, ROC (Receiver Operating Characteristic curve) is a probability curve and AUC (Area Under the Curve) represents the degree or measure of separability [14].
- Confusion Matrix: The confusion matrix is a table that describes the models' effectiveness and performance, on test subset where the true values are unknown. Its form seemed in the following table:



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 3.12: Confusion Matrix Table [10]

Where:

- True Positives (TP) is when the prediction is positive, and the real value is true
- False Positives (FP) is when the actual class of the data point was false and the predicted is true
- False Negatives (FN) is when the prediction is negative, and the real value is false
- True Negatives (TN) is when the actual class of the data point was fals and the predicted is also false

From these values we can later get the Accuracy of our model, which is the percentage of total items classified correctly, and the Recall or Sensitivity or TPR which is the number of items correctly identified as positive out of total true positives:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.4}$$

$$Recall = \frac{TP}{TP + TN} \tag{3.5}$$

- Regression Metrics: The most common metrics for evaluating predictions on regression machine learning problems are the following three [14]:
  - Mean Squared Error: which measures the mean squares of the errors between the actual values and those predicted by the model

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Yi - \hat{Y}i)^2$$

  - Mean Absolute Error: which measures the sum of the differences between the real and the predicted value divided by the sum of all observations

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \mid (Yi - \hat{Y}i) \mid$$

  - R2 metric: which provides an indication of how good a set of predictions fit to the actual values

Chapter 4

Datasets-Tools

4.1 Introduction

In this Chapter we will present all the information about the two data sets that are used in this thesis. Along with the software and the tools that are utilized to study and implement the problem.

4.2 Datasets

The datasets were drawn from UCI-Machine Learning Repository (Center for Machine Learning and Intelligent Systems). The UCI was founded firstly as an archive by David Aha and some graduate students at UC Irvine in 1987 and the current website is designed by Arthur Asuncion and David Newman in 2007. It contains a collection of datasets, databases and domain theories that can be explored and studied for machine learning [30].

More specifically, the first dataset, which called “Breast Cancer Wisconsin (Original)(WBC)” was obtained from the University of Wisconsin Hospital, USA by Dr. William H. Wolberg and contains cases of breast cancer. The samples from the clinical cases were collected periodically from dr. Wolberg and are assembled in eight groups. The total number of instances is 699 that refer to the rows, while the columns of the dataset are 11 and correspond to patient’s attributes. A detailed description of the dataset’s variables is provided in the next Table. The target value refers to whether the tumor is malignant or benign. The initial values were 2 for benign and 4 for malignant. Also, there are 16 instances that contain a single missing attribute and are replaced with the mean value of each column [31].

Attribute	Description
Sample code number	Patient id number
Clump Thickness	Integer (1-10)
Uniformity of Cell Size	Integer (1-10)
Uniformity of Cell Shape	Integer (1-10)
Marginal Adhesion	Integer (1-10)
Single Epithelial Cell Size	Integer (1-10)
Bare Nuclei	Integer (1-10)
Bland Chromatin	Integer (1-10)
Normal Nucleoli	Integer (1-10)
Mitoses	Integer (1-10)
Class	2 for benign, 4 for malignant

Meanwhile, by plotting our target variable “Class”, we identify the class distribution, with 458 of incidents benign and the rest 241 malignant.

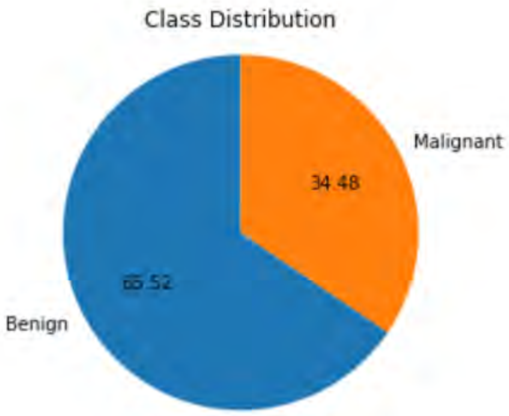


Figure 4.1: WBC Class Distribution

The second dataset, “Wisconsin Diagnostic Breast Cancer (WDBC)”, was also collected from the University of Wisconsin Hospitals, Madison by Dr. William H. Wolberg around 1995. The WDBC consists of 569 instances and has 32 patient’s attributes. Among these attributes there is one attribute that refers to patient’s id number, one that refers to the class (benign or malignant) and the rest 30 attributes consist of tumor diagnosis information, which are collected from 10 aspects and for each attribute three measure results are taken: mean, standard error and largest value. In order to create these 30 attributes Dr. Wolberg, used fluid samples, collected from patients with solid breast masses and a graphical computer program called Xcyt, which is competent of performing the analysis of cytological features based on a digital scan. The program uses a curve-fitting algorithm to compute the ten features from each one of the cells in the sample and returns a 30 real-valuated vector. So, for instance field 2 contains the mean of radius, field 12 the radius standard error and field 22 the radius largest value [12] [32]. The Table below shows the variable's information.

Attribute	Description
Sample code number	Patient id number
Radius	Mean of distances from center to points on the perimeter (integer)
Texture	Standard deviation of gray-scale values (integer)
Perimeter	Size of the core tumor (integer)
Area	Integer
Smoothness	Local variation in radius lengths (integer)
Compactness	Perimeter$^2/area - 1.0(integer)$
Concavity	Severity of concave portions of the contour (integer)
Concave points	Number of concave portions of the contour (integer)
Symmetry	Integer
Fractal dimension	”Coastline approximation” – 1 (integer)
Diagnosis	M=malignant and B=benign (character)

The class distribution of the previous data set is presented in the Figure 4.2.

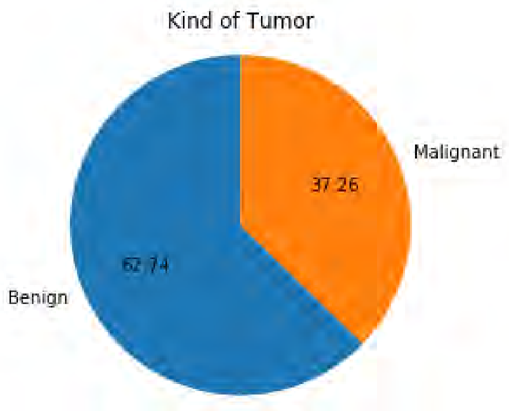


Figure 4.2: WDBC Class Distribution

4.3 Tools

This section contains a description of the working environment as well as the tools used to carry out the study.

4.3.1 Python

The whole code of this project has been written in Python. Python is a powerful programming language, which has several uses and is very common in the field of data science. For this reason, several platforms have been created to host this language and help in data analysis [33]. Some of the most common and powerful libraries for handling data structures are: Pandas, which contains a wide range of functions for studying data sets, NumPy, which includes everything from math computations to table management and Matplotlib, which is used to depict graphical representations.

4.3.2 Keras and TensorFlow

For Neural Network's implementation, an API (Application Programming Interface) is used, which is called "Keras". Keras is a very powerful API that is used mainly for the construction of neural networks. As an API we define intermediate software that allows communication between two applications. A prerequisite for Kera's operation is the existence of one of TensorFlow or Theano. In this thesis we focused on the use of TensorFlow, which is an open source library for fast numerical computing [27].

4.3.3 Anaconda

The platform we used to study and develop the models is called Anaconda. It is a free platform that contains various programs for mathematics, data science and engineering. The languages it hosts are Python and R. Also, it gives to users the ability to make an easy installation of the desirable version of python.

Chapter 5

Implementation

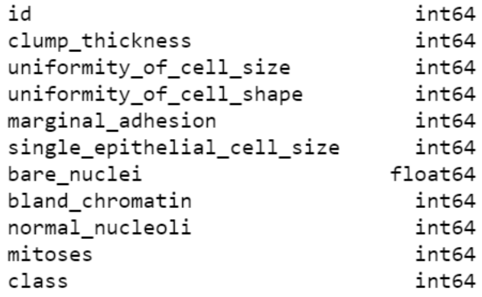
In this chapter we will present the methodology we applied to solve the problem of breast cancer prediction.

5.1 Data processing

Machine Learning is all about mathematical and statistical models with equations and numerical operations. That is, the input data to these algorithms should be numbers in order to work in a proper way. Therefore, the first level of the implementation is about the preparation of the two data sets.

5.1.1 Breast Cancer Wisconsin (Original)(WBC)

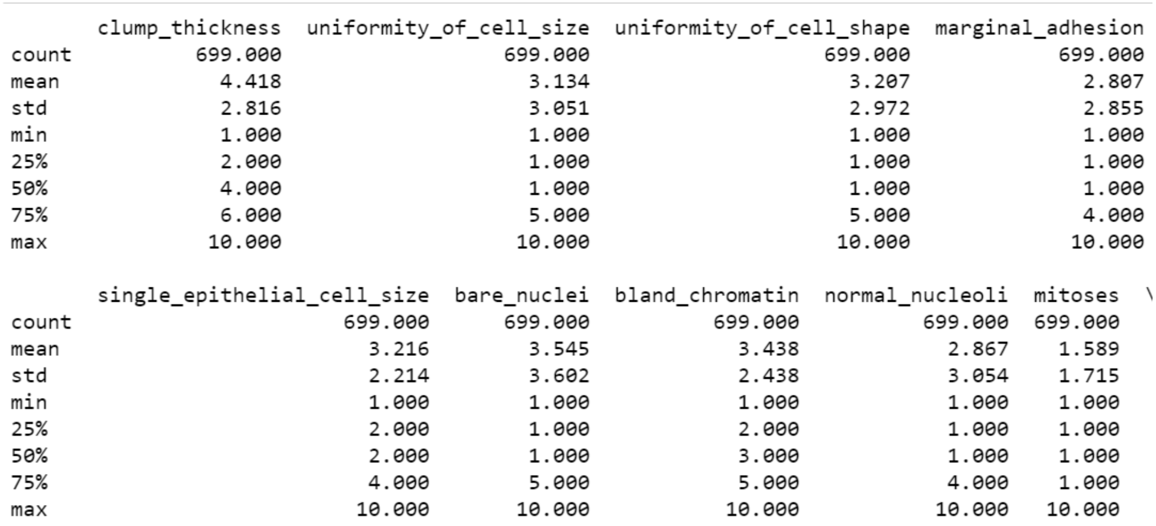
As mentioned earlier, in this thesis, we've tried to predict the breast cancer disease based on patients' medical information. In order to do so we've worked with two different datasets, WBC and WDBC. Firstly, using Pandas python's library, we cleaned the data from non-appropriate or duplicated values. All variables in WBC data set, as we can see below, are numbers so no changes needed to be made.



```
id                              int64
clump_thickness                 int64
uniformity_of_cell_size         int64
uniformity_of_cell_shape        int64
marginal_adhesion               int64
single_epithelial_cell_size     int64
bare_nuclei                   float64
bland_chromatin                 int64
normal_nucleoli                 int64
mitoses                         int64
class                           int64
```

Figure 5.1: Variable's type in WBC

Also, there are useful Panda's functions that return descriptive statistics and information about the variables of the dataset. The statistical summary of the features in the first dataset are given bellow:



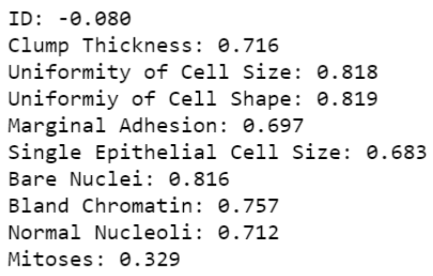
```
       clump_thickness  uniformity_of_cell_size  uniformity_of_cell_shape  marginal_adhesion
count          699.000                  699.000                   699.000            699.000
mean             4.418                    3.134                     3.207              2.807
std              2.816                    3.051                     2.972              2.855
min              1.000                    1.000                     1.000              1.000
25%              2.000                    1.000                     1.000              1.000
50%              4.000                    1.000                     1.000              1.000
75%              6.000                    5.000                     5.000              4.000
max             10.000                   10.000                    10.000             10.000

       single_epithelial_cell_size  bare_nuclei  bland_chromatin  normal_nucleoli  mitoses  \
count                      699.000      699.000          699.000          699.000  699.000
mean                         3.216        3.545            3.438            2.867    1.589
std                          2.214        3.602            2.438            3.054    1.715
min                          1.000        1.000            1.000            1.000    1.000
25%                          2.000        1.000            2.000            1.000    1.000
50%                          2.000        1.000            3.000            1.000    1.000
75%                          4.000        5.000            5.000            4.000    1.000
max                         10.000       10.000           10.000           10.000   10.000
```

Figure 5.2: Statistical Summary of the WBC Dataset

In the first data set, we changed the class values from 2 to 0 and from 4 to 1 to make it a binary problem.

By using the "corr" function it's easy to see the most useful variables for building the model since it gives us the correlations between variables. The Pearson factor, which gives us the correlation between two independent variables, was also used. In the next figure we can see the values of "corr" function.

It's easy to understand that the "ID" attribute, that contains the sample's code number, does not contribute to all at the classification and is not closely related to the target variable. Therefore, this variable was removed from our data set.



```
ID: -0.080
Clump Thickness: 0.716
Uniformity of Cell Size: 0.818
Uniformiy of Cell Shape: 0.819
Marginal Adhesion: 0.697
Single Epithelial Cell Size: 0.683
Bare Nuclei: 0.816
Bland Chromatin: 0.757
Normal Nucleoli: 0.712
Mitoses: 0.329
```

Figure 5.3: Results of “corr” function

Despite the fact that the data are in the same scale between 1-10, with “MinMaxScaler” function, which is a function from “Sklearn” Python’s library, all data variables received values from 0 to 1 for improved performance and more accurate results.

Feature selection and importance are procedures in Machine Learning and statistics, where a subset of the whole data set is chosen, to be used in the training and testing parts of the learning algorithm. Two of their benefits are the improvement of accuracy and the reduction of overfitting. “Extra Trees Classifier”, a bagged decision tree, is used to estimate the importance of features. The most important attributes of the data set are depicted in the following figure.

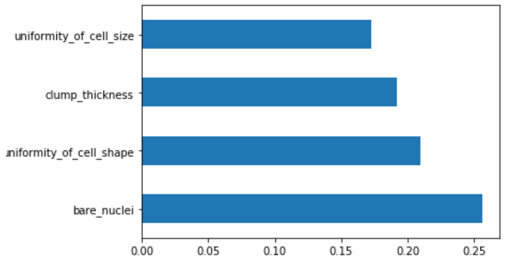


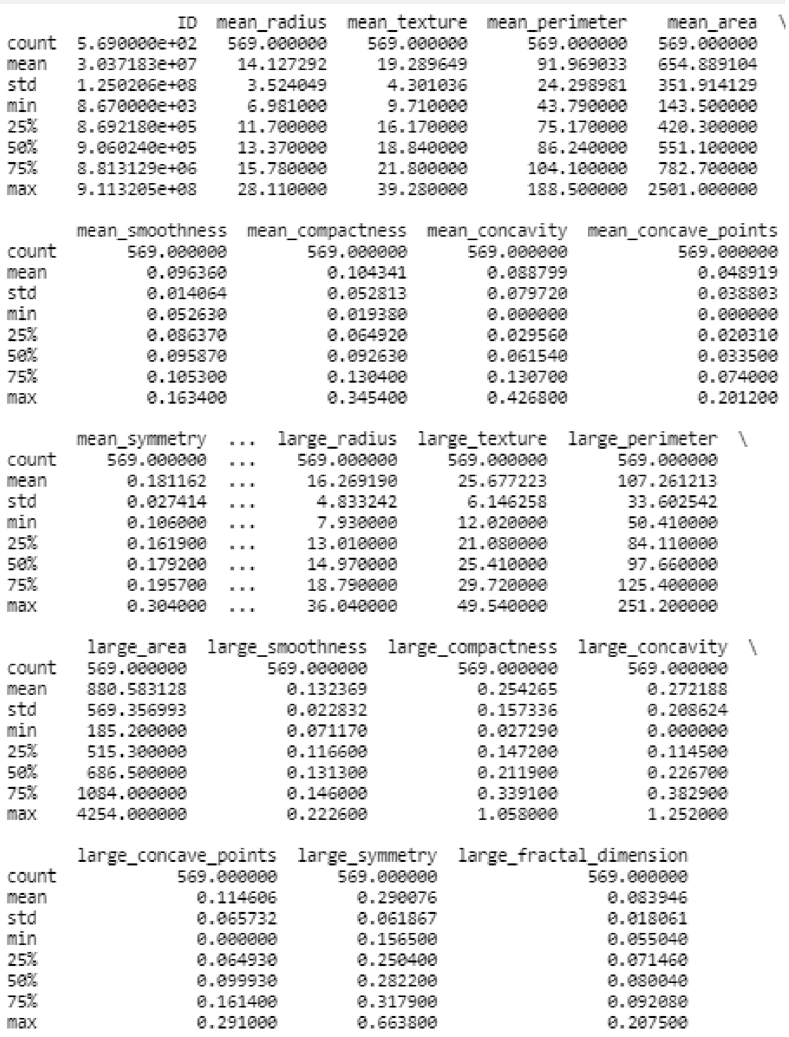
Figure 5.4: Feature Importance of WBC

Nevertheless, these results showed low accuracy, so we did not keep only these specific values to make predictions but the entire data set. Concluding the process of data processing, we divide the set of data into two parts: The set for training the algorithm and that for its control. As a percentage for both data sets, we chose 75% for education and the remaining 25% for control.

5.1.2 Wisconsin Diagnostic Breast Cancer (WDBC)

In both datasets, the same data pre-processing steps were applied. Firstly, we noticed that there were no missing values in the data cleaning process. In WDBC dataset our target value was categorical. We solved this problem by using the “One Hot Encoding”process . “One Hot Encoding” is a process by which categorical variables are converted into a form that could be provided to ML algorithms so that they could perform a better in prediction.

Some statistics also are depicted in the next figure.



```
                 ID  mean_radius  mean_texture  mean_perimeter    mean_area  \
count  5.690000e+02   569.000000    569.000000      569.000000   569.000000
mean   3.037183e+07    14.127292     19.289649       91.969033   654.889104
std    1.250206e+08     3.524049      4.301036       24.298981   351.914129
min    8.670000e+03     6.981000      9.710000       43.790000   143.500000
25%    8.692180e+05    11.700000     16.170000       75.170000   420.300000
50%    9.060240e+05    13.370000     18.840000       86.240000   551.100000
75%    8.813129e+06    15.780000     21.800000      104.100000   782.700000
max    9.113205e+08    28.110000     39.280000      188.500000  2501.000000

       mean_smoothness  mean_compactness  mean_concavity  mean_concave_points
count       569.000000        569.000000      569.000000           569.000000
mean          0.096360          0.104341        0.088799             0.048919
std           0.014064          0.052813        0.079720             0.038803
min           0.052630          0.019380        0.000000             0.000000
25%           0.086370          0.064920        0.029560             0.020310
50%           0.095870          0.092630        0.061540             0.033500
75%           0.105300          0.130400        0.130700             0.074000
max           0.163400          0.345400        0.426800             0.201200

       mean_symmetry  ...  large_radius  large_texture  large_perimeter  \
count     569.000000  ...    569.000000     569.000000       569.000000
mean        0.181162  ...     16.269190      25.677223       107.261213
std         0.027414  ...      4.833242       6.146258        33.602542
min         0.106000  ...      7.930000      12.020000        50.410000
25%         0.161900  ...     13.010000      21.080000        84.110000
50%         0.179200  ...     14.970000      25.410000        97.660000
75%         0.195700  ...     18.790000      29.720000       125.400000
max         0.304000  ...     36.040000      49.540000       251.200000

        large_area  large_smoothness  large_compactness  large_concavity  \
count   569.000000        569.000000         569.000000       569.000000
mean    880.583128          0.132369           0.254265         0.272188
std     569.356993          0.022832           0.157336         0.208624
min     185.200000          0.071170           0.027290         0.000000
25%     515.300000          0.116600           0.147200         0.114500
50%     686.500000          0.131300           0.211900         0.226700
75%    1084.000000          0.146000           0.339100         0.382900
max    4254.000000          0.222600           1.058000         1.252000

       large_concave_points  large_symmetry  large_fractal_dimension
count            569.000000      569.000000               569.000000
mean               0.114606        0.290076                 0.083946
std                0.065732        0.061867                 0.018061
min                0.000000        0.156500                 0.055040
25%                0.064930        0.250400                 0.071460
50%                0.099930        0.282200                 0.080040
75%                0.161400        0.317900                 0.092080
max                0.291000        0.663800                 0.207500
```

Figure 5.5: WDBC Statistics

Similar to the first data set, “corr” function suggests removing the “ID” attribute, which contains the patient’s code number, from the dataset due to low correlation to the target variable.

The Figure 5.6 shows the correlation matrix results among all attributes.

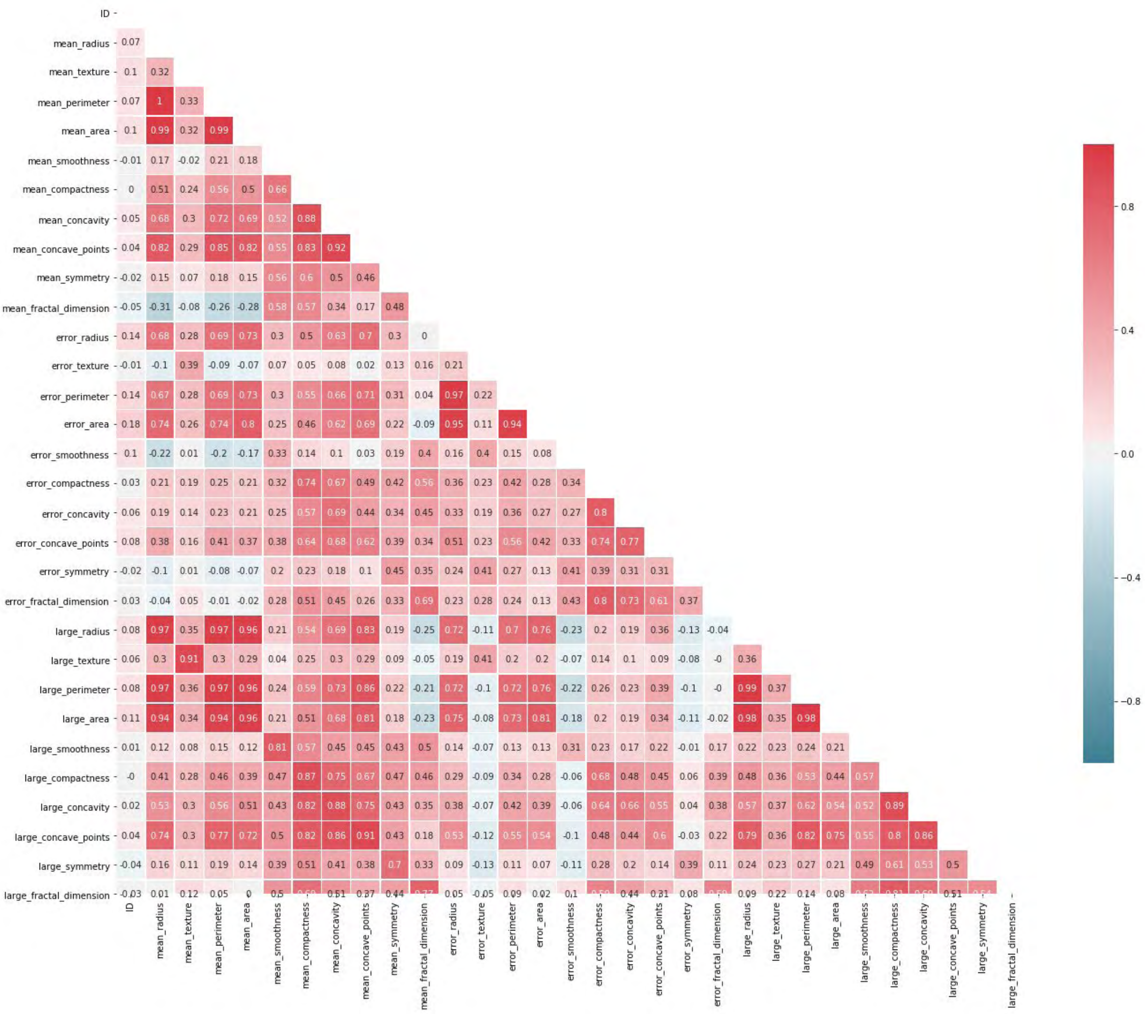


Figure 5.6: WDBC-Correlation Matrix “heatmap”

Looking at the matrix the aim is to understand how the 30 characteristics of the data set are related to each other. We can instantly verify the presence of multicollinearity between some of our variables. By calculating correlations between variables, we can observe that some of them are notably correlated. For instance the “mean_radius” column has a correlation of 1 and 0.99 with “mean_perimeter” and “mean_area” columns, respectively. This is probably because these three columns essentially contain the same or even redundant information. Also, “mean_radius” has a correlation of 0.97 with the “large_radius” column. In fact, each of the 10 key attributes display very high correlations between its ”mean” and ”large” columns. This maybe explained, because the “large” values are also the mean values of the three largest values amid all instances. Therefore, a good idea is to remove the “large” columns of the dataset and emphasize more on “mean” columns. However, the results didn’t show any improvement in accuracy.

Following, we converted data that are in different order of magnitude, in order for the process to work properly and we used the same scale. For this purpose, we used the “StandardScaler” function, which is a “Sklearn” function that aims to convert data on the same scale for better algorithm execution. While applying StandardScaler, each feature of your data should be normally distributed so that it would scale the distribution to a mean of zero and a standard deviation of one.

Too many features cost more computational time and memory, which is also known as the “Curse of Dimensionality”. In this part PCA was implemented to convert the 30-dimension data set into a lower dimensional space. Ten principal components are decided to be the best choice for WDBC data set, based on 95% of total correlation. Obviously, different number of components are decided based on different criteria.

Concluding, the data set was separated into a train set of 80% and a test set of 20% for education and control respectively.

5.2 Prediction Models

This subsection presents the algorithms used to build the prediction model.

5.2.1 Logistic Regression

The first model was designed with the Logistic Regression algorithm from “Sklearn” library. In this model, the probabilities relating the possible outcomes of a single trial are demonstrated using a logistic function. The first parameter was the penalty, which is used to specify the norm used in the penalization. Here this value is set by default with “l2”. The next parameter was tolerance for stopping criteria and took the value: $1e^{-4}$. The “C” parameter is an inverse regularization parameter with the default value of 1.0.

Table 5.1: Logistic Regression Parameters

Parameter	Value
Penalty	l2
tol	$1e^{-4}$
c	1.0

5.2.2 KNN

The next model was implemented with K-Nearest Neighbors. Here it’s decided to be used 5 number of neighbors, as the first parameter. The “metric” parameter also affects the efficiency of the model and took the value “minkowski”. This parameter together with the power parameter “p”, where p=2, is equivalent to the standard Euclidean metric.

Table 5.2: KNN Parameters

Parameter	Value
n_neighbors	5
metric	Minkowski
p	2

5.2.3 SVM

The next model was built using Support Vector Machine. In this model, the first setting parameter is cost C, which determines how strict we want to be with the errors. For our model the value was chosen 1.0. Next parameter is the kernel option that for our model the Radial Basis Function (RBF) was used. And finally, Epsilon (e) which defines the terms that Hyperplane has and as a value was chosen 0.1. The SVR execution function came from the Sklearn Library of Python.

Table 5.3: SVM Parameters

Parameter	Value
Kernel	RBF
c	1.0
epsilon	0.1

5.2.4 Naïve Bayes

This model used Gaussian Naïve Bayes from “Sklearn” Library. The parameters were set by default. With the “.fit” method, the algorithm applied according to “x_train” array into “y_train” target values.

Table 5.4: NB Parameters

Parameter	Value
priors	None
var_smoothing	$1e^{-9}$

5.2.5 Random Forest

The functions of the “Sklearn” Library of Python were used to build this model, as well. Random forest is an ensemble machine learning algorithm. The main parameters that affect the model are the number of trees in the forest (n_estimators) and the function to measure the quality of a split. In this last measure we chose, “entropy” value for information gain.

Table 5.5: Random Forest Parameters

Parameter	Value
n_estimators	10
criterion	Entropy
random state	0

5.2.6 MLP

For the last model, neural networks were used and specifically the Perceptron or MLP multilevel network. The configuration parameters of this model were several, but the ones that played the biggest role were the number of hidden layers, the core functions used and finally the activation functions.Once again, these parameters were selected by testing the available parameter values that we had for classification problems.

Additionally, we tried to increase the number of hidden layers but, by observing the price of RMSE, we saw that its reduction was not significant, while at the same time the time and complexity of the model increased. An attempt was also made to change the activation function, but again the results were not as good as the ones we finally used (“relu”).

Table 5.6: MLP Parameters

Parameter	Value
Hidden Layers	2
Activation Function	relu
Kernel Function	uniform
Units	5 for WBC and 9 for WDBC

The activation function for the output layer was “sigmoid” for both data sets.

Finally, in order to achieve the best possible result through this model, an optimization function, called “adam” was used. This function is used quite often in classification problems. ”Adam” optimization is a stochastic gradient descent method that is rely on adaptive estimation of first-order and second-order moments.

5.3 Evaluation Metrics

While applying all algorithms in order to create the models , the following metrics have been used to evaluate their results.

- Accuracy: is the ratio of the correctly predicted instances to the total number of instances .
- Confusion Matrix: is a summary of prediction results on our classification problem. All the predictions, nonetheless if they are correct or incorrect are summarized with count values for each class, which is the key to the confusion matrix. It provides a vision not only into the errors being made but also into the importance of the types of these errors. Therefore, it’s very important in breast cancer prediction, because even the cases that predicted wrongly that someone was not sick while he was actually sick appear.

Finally, a 10-fold cross-validation test harness is used to demonstrate each metric when applying different algorithm evaluation metrics. This function is the cross_val_score used to report the performance from “Sklearn” Library.

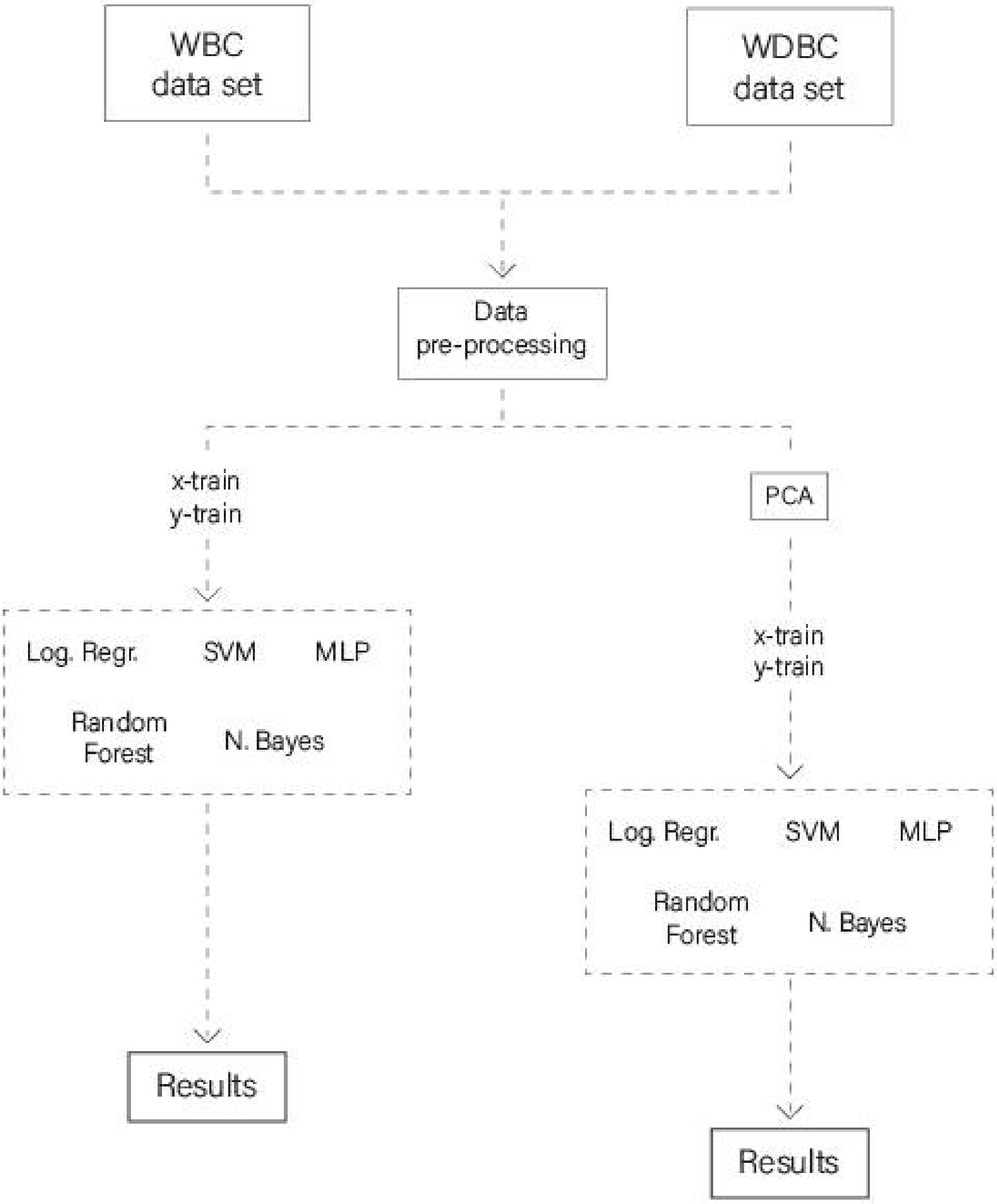


Figure 5.7: Methodology development of the thesis

Chapter 6

Results

In this chapter, the results of the models we built, after being trained and tested in the two datasets, are reported. According to the above, in order to apply our classifiers and evaluate them, we applied the 10-fold cross validation test, a technique used in evaluating predictive models that splits the original set into a training sample to train the model, and a test set to evaluate it.

Initially, as mentioned above, the predictive models that have been implemented predict whether a patient's breast tumor is benign or malignant. For each of the data sets we will present the results of the models we built.

6.1 Machine Learning

Logistic Regression

The first model was implemented using the Logistic Regression algorithm. The results for each of the two data sets are depicted in the following table and as we can see, the values of accuracy are very high. For the first data set we have a success rate close to 96% while in the second this percentage is close to 97%.

Table 6.1: Logistic Regression Accuracy Results

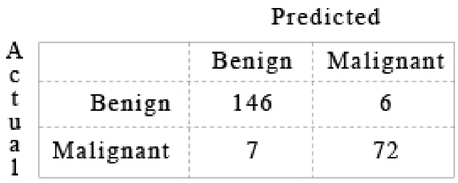
Dataset	Accuracy
WBC	96,3%
WDBC	97,3%

The very good results can also be reaffirmed with the classification metrics of precision, recall and F1. More specifically, in WBC dataset we have a total precision of 0.914, 0.95 for benign predictions and 0.92 for malignant ones. Precision is the proportion of positive identifications that were actually correct. Recall is the proportion of actual positives that were identified correctly. So, for WBC our model has correctly identified 96% of all benign tumors and 91% of malignant tumors. F1 score is the harmonic mean of precision and recall.

Table 6.2: Logistic Regression Results

Dataset	Precision	Recall	F1 score
WBC	91,4%	94%	95%
WDBC	96,2%	94,3%	96,5%

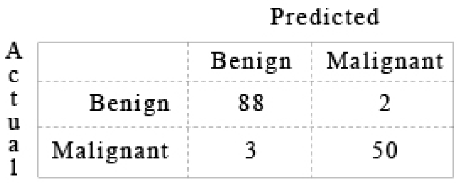
The last classification metric that we used to evaluate our model, is confusion matrix. According to the above, with confusion matrix we can see all types of important errors. As we can see, in WDBC the values of false predictions are very low. This implies that wrong predictions as , for example, that the tumor was benign but actually was malignant, were only 2. The following Figures and Tables represent the results of confusion matrix for the two datasets.



Predicted

		Benign	Malignant
Actual	Benign	146	6
	Malignant	7	72

Figure 6.1: WBC-Logistic Regression Confusion Matrix



Predicted

		Benign	Malignant
Actual	Benign	88	2
	Malignant	3	50

Figure 6.2: WDBC-Logistic Regression Confusion Matrix

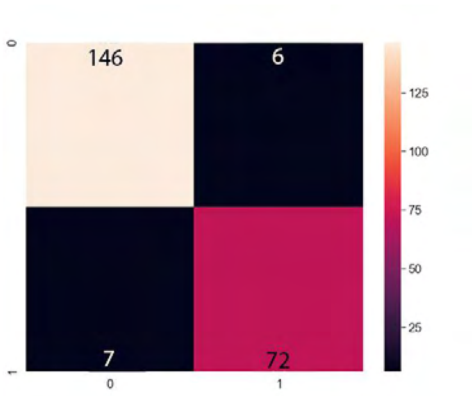


Figure 6.3: WBC-Logistic Regression Confusion Matrix "heatmap"

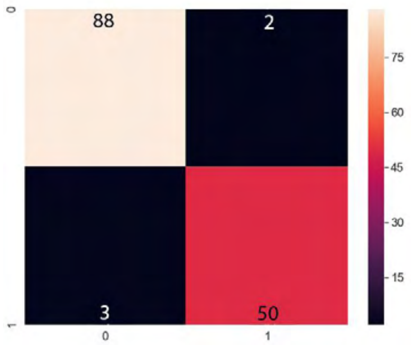


Figure 6.4: WDBC-Logistic Regression Confusion Matrix "heatmap"

KNN

The second model that was implemented used the K-Nearest Neighbor classification algorithm. The number of neighbors we selected were 5. The results here are also very high with accuracy more than 95% for both data sets. The accuracy is represented below:

To go even deeper into the performance of this model we used again the following classification metrics :

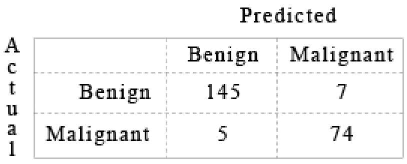
Table 6.3: KNN Accuracy Results

Dataset	Accuracy
WBC	96,6%
WDBC	97,67%

Table 6.4: KNN Results

Dataset	Precision	Recall	F1 score
WBC	92,3%	93%	94,8%
WDBC	97%	88,7%	95%

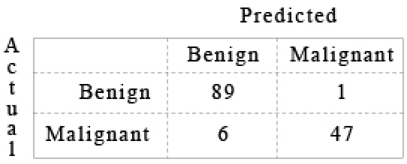
In the next Figures and Tables, the results of Confusion Matrix are represented. In WBC, as we can see the occasions that the patient was wrongly predicted as healthy, are only 7 and the occasions that the tumor was malignant, but we predicted it as benign were 5. Also, in case of WDBC, these values are extremely low with 1 and 6 occasions respectively. These are the most important errors for this classification problem and its very significant to keep these values very low, so that people can prevent cancer or can prevent themselves from taking unnecessary treatments in case they are healthy.



Predicted

Actual		Benign	Malignant
	Benign	145	7
	Malignant	5	74

Figure 6.5: WBC-KNN Confusion Matrix



Predicted

Actual		Benign	Malignant
	Benign	89	1
	Malignant	6	47

Figure 6.6: WDBC-KNN Confusion Matrix

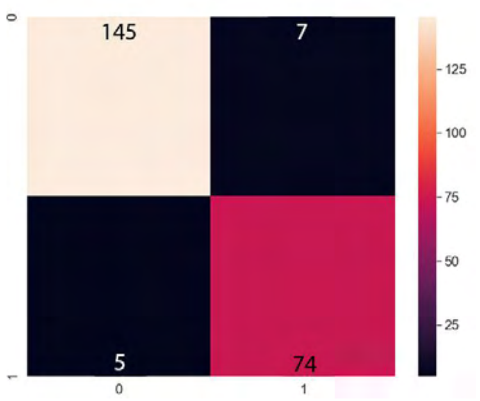


Figure 6.7: WBC-KNN Confusion Matrix “heatmap”

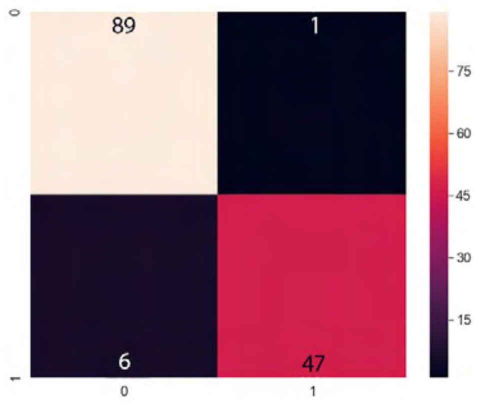


Figure 6.8: WDBC-KNN Confusion Matrix “heatmap”

Naïve Bayes

The following model was built with the Naïve Bayes algorithm. The accuracy presented was 0.96 and 0.926 points for WBC and WDBC data sets respectively, using 10 k-folds cross validation.

Table 6.5: Naïve Bayes Accuracy Results

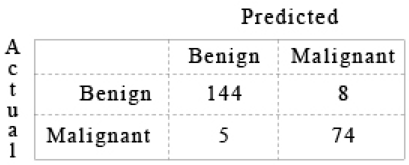
Dataset	Accuracy
WBC	96%
WDBC	92,6%

The rest classification metrics are depicted in the following Table. 90% precision in the WBC means that when the model predicts a malignant tumor , it is correct 90% of the time. The results of WDBC are not so good, like in previous models

Table 6.6: NB Results

Dataset	Precision	Recall	F1 score
WBC	90,2%	93,7%	94,8%
WDBC	88,7%	88%	91%

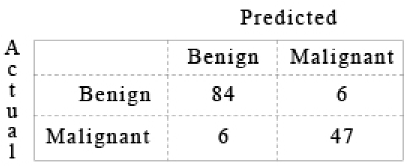
The following Figures and Tables show the Confusion Matrix results for both WBC and WDBC data sets. In the second data set, the results of confusion matrix for the false predictions remain low, however the previous models presented better values .



Predicted

		Benign	Malignant
Actual	Benign	144	8
	Malignant	5	74

Figure 6.9: WBC-NB Confusion Matrix



Predicted

		Benign	Malignant
Actual	Benign	84	6
	Malignant	6	47

Figure 6.10: WDBC-NB Confusion Matrix

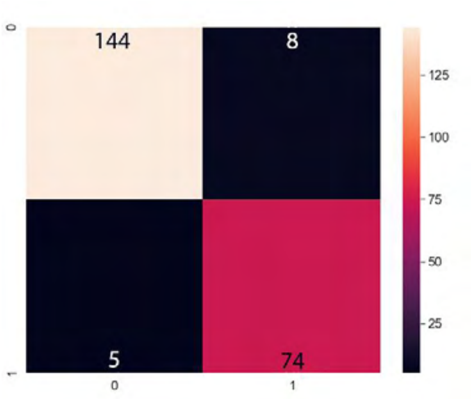


Figure 6.11: WBC-NB Confusion Matrix “heatmap”

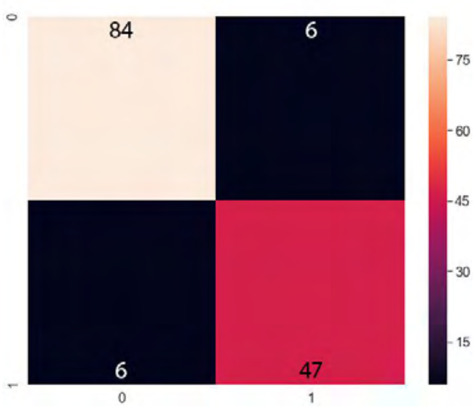


Figure 6.12: WDBC-NB Confusion Matrix “heatmap”

SVM

One promising model was implemented using Support Vector Machine algorithm. In Table 6.7 we can see SVM’s algorithm very good performance for both data sets

Table 6.7: SVM Accuracy Results

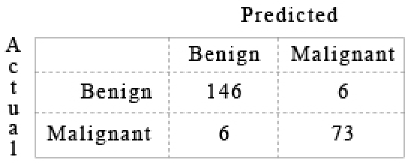
Dataset	Accuracy
WBC	96,8%
WDBC	97,5%

The rest classification metrics are depicted in the following Table. The results of WDBC are quite the same with those of Logistic Regression model.

Table 6.8: SVM Results

Dataset	Precision	Recall	F1 score
WBC	93%	93%	95%
WDBC	96,3%	94,4%	96,5%

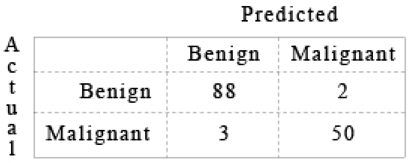
In order to see the results of our model, we used the Confusion Matrix in more detail, as before, because it gives us the opportunity to evaluate in depth even the wrong predictions, that is, the important errors. For WBC the results are very good, because of the high values in the correct predictions and at the same time the very low value of 6 points in the wrong predictions.



Predicted

		Benign	Malignant
Actual	Benign	146	6
	Malignant	6	73

Figure 6.13: WBC-SVM Confusion Matrix



Predicted

		Benign	Malignant
Actual	Benign	88	2
	Malignant	3	50

Figure 6.14: WDBC-SVM Confusion Matrix

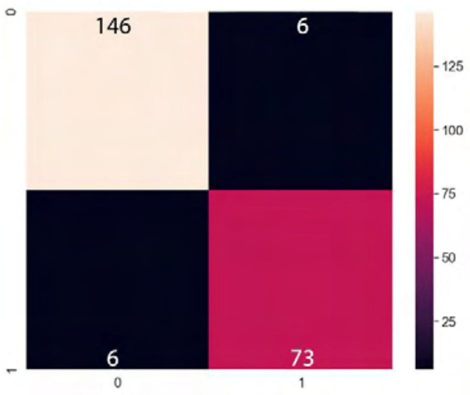


Figure 6.15: WBC-SVM Confusion Matrix “heatmap”

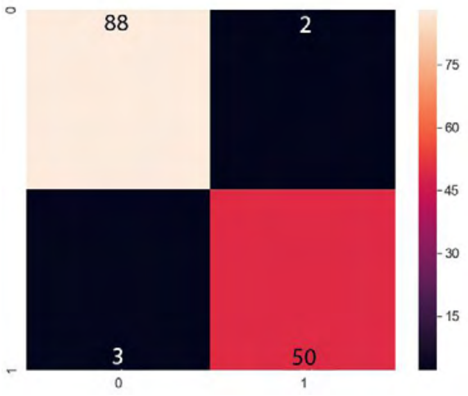


Figure 6.16: WDBC-SVM Confusion Matrix “heatmap”

Random Forest

Using the Random Forest algorithm, the results are also good, especially for the second WDBC data set. The accuracy results as we can see in the next Table are relatively close to those of the previous models.

Table 6.9: Random Forest Accuracy Results

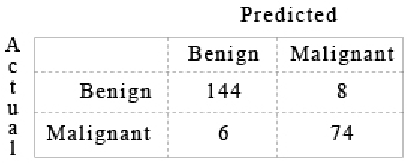
Dataset	Accuracy
WBC	95,7%
WDBC	96%

The evaluation metrics of precision, recall and F1 score are represented in the Table 6.10.

Table 6.10: Random Forest Results

Dataset	Precision	Recall	F1 score
WBC	91,5%	94,9%	94,8%
WDBC	97%	98%	97,7%

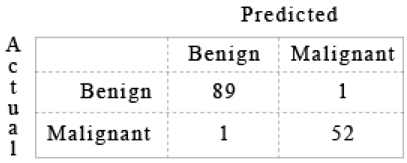
Subsequently, Confusion Matrix's results were also very good. However, in WBC the values of wrong predictions are bigger than those of other models, so that makes this particular model less preferable. On the other hand, in WDBC these values are extremely low.



Predicted

Actual		Benign	Malignant
	Benign	144	8
	Malignant	6	74

Figure 6.17: WBC-Random Forest Confusion Matrix



Predicted

Actual		Benign	Malignant
	Benign	89	1
	Malignant	1	52

Figure 6.18: WDBC-Random Forest Confusion Matrix

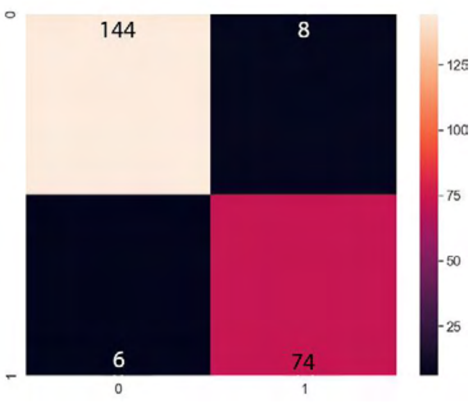


Figure 6.19: WBC-Random Forest Confusion Matrix “heatmap”

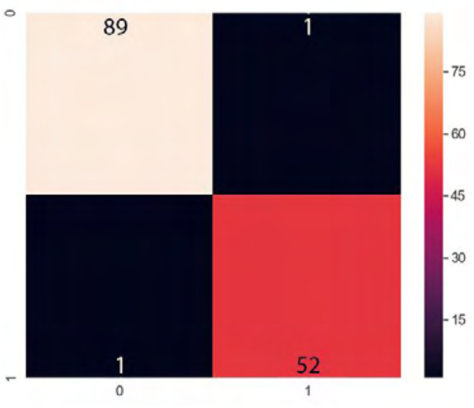


Figure 6.20: WDBC-Random Forest Confusion Matrix “heatmap”

Machine Learning Algorithm Comparison

For both data sets, all machine learning algorithms except from Neural Networks are depicted in the following Figures.

For WBC it’s clear that almost all algorithms have very high accuracy. Logistic Regression algorithm has the biggest value range between 0.93 to approximately 0.97 points, while Naïve Bayes has the smallest range between 0.955 to 0.965. The SVM has also a big range among the values but also presents a better performance.

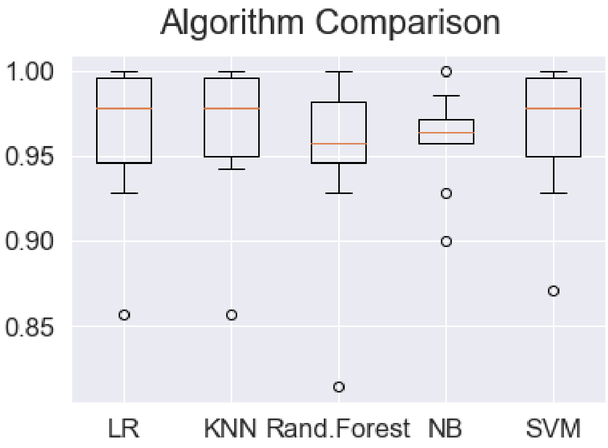


Figure 6.21: WBC Algorithm Comparison

In WDBC, among all algorithms, SVM has again the better performance. Logistic Regression and KNN algorithms have a small value range.

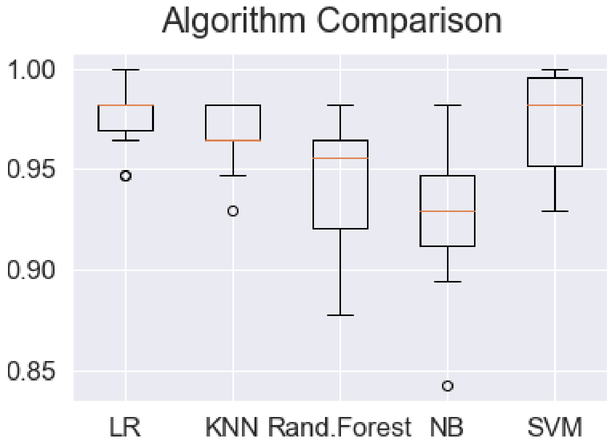


Figure 6.22: WDBC Algorithm Comparison

MLP Neural Network

The last model we used was the one using MLP Neural Networks. The classification accuracy results are very high which implies a good performance of this model. In WBC we have a result close to 0.95 and in WDBC close to 0.99.

Table 6.11: MLP Accuracy Results

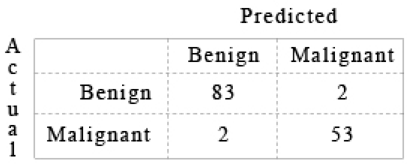
Dataset	Accuracy
WBC	98%
WDBC	98,5%

The good performance of this model is depicted also in the Table 6.12.

Table 6.12: MLP Results

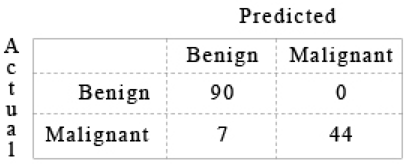
Dataset	Precision	Recall	F1 score
WBC	95%	98,3%	97,5%
WDBC	97,5%	94%	97,5%

The model's good performance can also be proved through the Confusion Matrix. It is noteworthy that in WDBC we have no case where we predicted malignancies while the tumor was actually benign.



		Predicted	
Actual		Benign	Malignant
	Benign	83	2
	Malignant	2	53

Figure 6.23: WBC-MLP Confusion Matrix



		Predicted	
Actual		Benign	Malignant
	Benign	90	0
	Malignant	7	44

Figure 6.24: WDBC-MLP Confusion Matrix “heatmap”

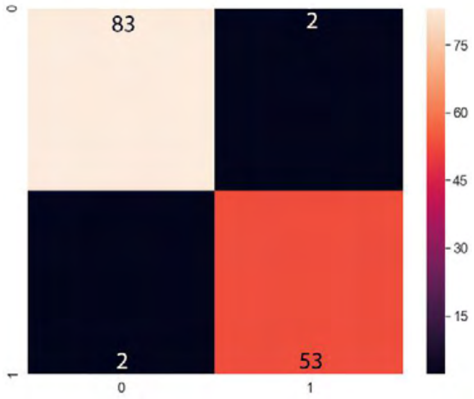


Figure 6.25: WBC-MLP Confusion Matrix “heatmap”

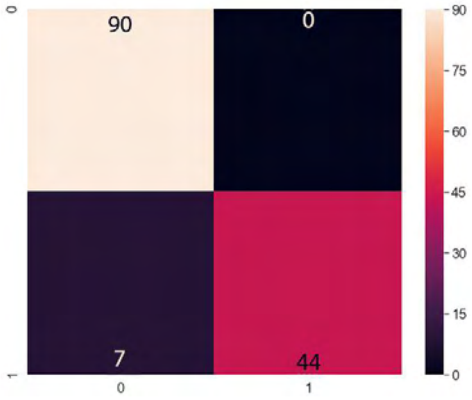


Figure 6.26: WDBC-MLP Confusion Matrix “heatmap”

MLP Neural Networks is certainly a very effective model with high accuracy and good performance.

Chapter 7

Conclusion

7.1 Conclusions and Discussion

To analyze medical data, various data mining and machine learning methods are available. An important challenge in data mining and machine learning areas is to build accurate and computationally efficient classifiers for Medical applications. In this thesis we developed and compared to each other, six different machine learning algorithms, Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Naïve Bayes, Random Forest and MLP Neural Networks. In addition, we applied feature selection techniques as well as dimensionality reduction methods, to exploit the data in the most efficient way. The evaluation metrics that are used are Accuracy, Precision, Recall, F1 score and Confusion Matrix. After a step by step analysis on the results of these metrics, Neural Networks outperformed the other used classifiers. More specifically, in WBC dataset MLP's and SVM's performance was very satisfying. However, MLP has better results and fewer errors according to confusion matrix. In the second dataset, WDBC, almost the same results were reached. MLP reaches the highest accuracy of 98,5%. Also, SVM and Random Forest algorithms presented very good results. PCA, as a dimensionality reduction method, manifests some advantages in model's performance and efficiency, using at the same time k-folds cross validation. So, the combination of these algorithms reaches better results, as a hybrid model. The results of this thesis were in line with the overall conclusion of the previous studies mentioned in Chapter 2. Neural Networks and SVM algorithms performed, most of the times, better accuracy and performance.

Concluding, , breast cancer prediction can be achieved with high accuracy and precision through those models and help medical world prevent and detect cancer. However, since there is still the possibility of an error, neither doctors nor patients can undoubtedly rely on these predictions for the time being. Combining these models with medical examinations can lead to more accurate diagnosis .

7.2 Future Work

As for future extensions regarding this thesis, both data and prediction models can be involved. Firstly, it we would of high significance to be able to extract more data and test these algorithms in other data sets. It is still an ideal scenario if we could extract more recent data from hospitals, process them from scratch and bring them to a form that can be analyzed. This would give us the opportunity to have a better insight of the efficiency and effectiveness of the cancer prediction models and be able to help medical research.

Regarding prediction models, more data would certainly drive researchers to experiment and introduce new, even more efficient models . Because the accuracy of our models was already very high, in order to improve it even more, it would be interesting if new, hybrid models could be created . As for PCA as a linear method it converts feature space into uncorrelated variables based on linear functions. In terms of nonlinear feature reduction methods, some other techniques can also be tested. Finally, the goal is to make a very strong prediction model that can safely help in cancer prediction and detection.

Bibliography

[1] National institute of health. https://seer.cancer.gov/statfacts/html/breast.html. Accessed: 31-03-2020.

[2] Cross-validation. http://genome.tugraz.at/proclassify/help/pages/XV.html. Accessed: 31-03-2020.

[3] Overfitting and underfitting. https://mc.ai/overfitting-and-underfitting-bug-in-ml-models/. Accessed: 31-03-2020.

[4] K nearest neighbors- classification. https://www.saedsayad.com/k_nearest_neighbors.html. Accessed: 01-04-2020.

[5] M. He et al. K nearest gaussian: A model fusion based framework for imbalanced classification with noisy dataset. *Artificial Intelligence Research*, 4(2):126–135, Aug 2015. doi: 10.5430/air.v4n2p126.

[6] Support vector machine: Introduction to machine learning algorithms. https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47. Accessed: 03-04-2020.

[7] Support vector machine - regression. https://www.saedsayad.com/support_vector_machine_reg.htm. Accessed: 03-04-2020.

[8] Random forest simple explanation. https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d, note = Accessed: 22-04-2020.

[9] Understanding of multilayer perceptron(mlp). `https://medium.com/@AI_with_Kain/understanding-of-multilayer-perceptron-mlp-8f179c4a135f`. Accessed: 04-04-2020.

[10] Understanding confusion matrix. `https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62`. Accessed: 01-04-2020.

[11] Breast cancer facts figures 2019-2020, american cancer society, atlanta: American cancer society. `chrome-extension://oemmndcbldboiebfnladdacbdfmadadm/https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/breast-cancer-facts-and-figures/breast-cancer-facts-and-figures-2019-2020.pdf`. 2019, Accessed: 31-03-2020.

[12] H. Wang and S. W. Yoon. Breast cancer prediction using data mining method. In *Proceedings of the 2015 Industrial and Systems Engineering Research Conference*, pages 818–828, Nashville, TN, USA, May-June 2015.

[13] Georgouli A. Τεχνητή νοημοσύνη. In *Μηχανική Μάθηση*, chapter 4. Kallipos publications, Athens, 2015.

[14] J. Brownlee. *Machine Learning Mastery with Python*. Melbourne Australia, 2008.

[15] Overfitting and underfitting with machine learning algorithms. `https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/`. Accessed: 31-03-2020.

[16] Gradient descent training with logistic regression. `https://towardsdatascience.com/gradient-descent-training-with-logistic-regression-c5516f5344f7`. Accessed: 01-04-2020.

[17] Logistic regression: Detailed overview. `https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc`. Accessed: 01-04-2020.

[18] Logistic regression for machine learning. https://machinelearningmastery.com/logistic-regression-for-machine-learning/. Accessed: 01-04-2020.

[19] Support vector machines for machine learning. https://machinelearningmastery.com/support-vector-machines-for-machine-learning/. Accessed: 03-04-2020.

[20] Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, and Vipin Kumar. *Introduction to Data Mining (2nd Edition)*. Pearson, 2nd edition, 2018.

[21] Support vector machine: Kernel trick; mercer's theorem. https://towardsdatascience.com/understanding-support-vector-machine-part-2-kernel-trick-mercers-theorem-e1e6848c6c4d. Accessed: 03-04-2020.

[22] A. Liaw M. Wiener. Classification and regression by randomforest. *R News*, 2/3(2):18–22, 2014. Available at:https://cogns.northwestern.edu/cbmg/LiawAndWiener2002.pdf.

[23] Understanding random forest. https://towardsdatascience.com/understanding-random-forest-58381e0602d2, note = Accessed: 22-04-2020.

[24] Everything you did and didn't know about pca. http://alexhwilliams.info/itsneuronalblog/2016/03/27/pca/?fbclid=IwAR2vjFVSQOCHteROsVUy_pjIWLEtMxU4Wu5HCz30MPkZmK4IidAtMUd3Zgg. Accessed: 21-04-2020.

[25] Dimensionality reduction: Does pca really improves classification outcome? https://towardsdatascience.com/dimensionality-reduction-does-pca-really-improve-classification-outcome-6e9ba21f0a32, note = Accessed: 21-04-2020.

[26] Introduction to artificial neural networks(ann). https://towardsdatascience.com/introduction-to-artificial-neural-networks-ann-1aea15775ef9. Accessed: 03-04-2020.

[27] J. Brownlee. *Develop Deep Learning Models on Theano and TensorFlow Using Keras*. Deep Learning with Python in Machine Learning Mastery, Melbourne Australia, 2018.

[28] Understanding activation functions in neural networks. `https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0`. Accessed: 03-04-2020.

[29] Y. Bengio A. Courvilee I. Goodfellow. *Deep Learning*. The MIT Press, Cambridge, 2016.

[30] Uci-machine learning repository. `http://archive.ics.uci.edu/ml/about.html`. Accessed: 05-04-2020.

[31] Original breast cancer wisconsin dataset. `http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Original)`. Accessed: 06-04-2020.

[32] Diagnostic breast cancer wisconsin dataset. `http://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic)`. Accessed: 06-04-2020.

[33] The python tutorial. `https://docs.python.org/3/tutorial/index.html`. Accessed: 05-04-2020.