Università degli Studi di Padova

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Triennale in
Ingegneria Informatica

Heart disease detection based on machine learning algorithms

Supervisor:
Prof. Gloria Beraldo

Student:
Elisa Borella
2007963

Academic Year 2022/2023
Graduation Date 29/09/2023

To my grandfather

Abstract

Cardiovascular diseases are the first cause of death all over the world. By using artificial intelligence algorithms and, in particular, machine learning approaches it is possible to predict risky situations due to heart disease. Various approaches are investigated in this thesis such as Neural Network, Support Vector Machine, Decision Tree, Naive Bayes, Logistic Regression and Stochastic Gradient Descent to extract predictive models in order to test for the presence or absence of heart disease.

Thanks to the public dataset from UCI, it is possible to take advantage of medical data to train the proposed models.

A comparison among the different approaches based on the performance was included in this thesis.

The tests of the proposed models revealed performances in terms of accuracy in the range 77%-90.6%. The Naive Bayes model has been the model with the highest accuracy (90.6%), highest precision (96.4%) and shortest time for classification (0.003 seconds).

Contents

Chapter 1

Introduction

This chapter presents the reasons that led to the writing of this thesis, the purposes of this work and this thesis structure. Section 1.2 is dedicated to present related works about heart disease detection with machine learning.

1.1 Motivations

The World Health Organization declares that cardiovascular diseases (CVDs) are the main cause of death in the whole world, having caused 17.9 million deaths in 2019[1]. This type of diseases comprehend coronary heart disease, cerebrovascular disease, rheumatic heart disease and others. Most of them could be prevented by avoiding risk behaviour, such as abuse of alcohol, tobacco, unhealthy food. The effects of CVDs might appear in the human body via high blood pressure, high blood glucose, high blood lipids. It is crucial to detect these pathologies in time, giving appropriate treatment to the patients with the aim of avoiding premature deaths. Artificial Intelligence can be helpful to identify which patients are at high risk of cardiovascular events, by simply performing some non-invasive tests such as blood test and electrocardiogram. A big amount of medical data is produced by Hospitals and it is necessary to find modern approaches to take advantage of it. To achieve the goal of detecting high-risk patients, it is possible to use many of the public datasets containing patient health information.

[1] https://www.who.int/health-topics/cardiovascular-diseases

1.2 Related work

Since CVDs have a great impact on our society, a lot of research works have been performed with the aim of finding techniques which could detect and predict this type of diseases with high accuracy. Most of them are based on machine learning models and a performance analysis summary of the ones consulted for this thesis is presented in Table 1.1.

Work	**Algorithm**	**Accuracy**
Kavitha et al. (2021) [1]	Decision Tree and Random Forest	88%
Dun et al. (2008) [2]	Neural Network	78%
Obaloluwa Olaniyi et al. (2015) [3]	Neural Network	85%
Obaloluwa Olaniyi et al. (2015) [3]	Support Vector Machine	87.5%
Sayad et al. (2014) [4]	Neural Network	88%
Chitra et al. (2013) [5]	Neural Network	85%
Priscila et al. (2017) [6]	Neural Network	90%
Miranda et al. (2021) [7]	Logistic Regression	91.67%
Miranda et al. (2021) [7]	Stochastic Gradient Descent	80%
Ramesh et al. (2022) [8]	Naive Bayes	86%

Table 1.1: Performance of related works

The most widely used approaches include Neural Networks, Support Vector Machine, Logistic Regression, Naive Bayes and Decision Tree. All the authors of the above texts decided to use the UCI dataset, also exploited in this thesis. In this thesis, all of them have been implemented and analyzed using Python programming.

In research work [8], the researchers proposed different approaches to detect CDVs using UCI dataset. In particular, Tree Classifier, Random Forest, and K-Nearest Neighbors are tested.

In work [1], a novel algorithm has been proposed. Various approaches have been analyzed such as Decision Tree (DT), Random Forest (RF) and a hybrid model (of DT and RF). The hybrid model achieved better accuracy than the two single models. In particular, the posterior probabilities in output from the Random Forest using the original dataset are provided in input to the Decision Tree algo-

rithm. Also, the researches have designed an application to perform heart disease prediction, giving the results of health examinations as input. The graphical user interface has been developed using Python's Tkinter library.

In [7], the researches applied Logistic Regression and Stochastic Gradient Descent to predict heart disease. The approach that gave the best results in terms of accuracy, precision and recall is Logistic Regression. Authors stated that having a larger dataset could lead to better results and that the small size of the dataset has been a limitation for their models performance. The algorithms proposed by the authors may constitute a non-invasive diagnosis tool to detect heart disease presence.

In the research [3], the researchers developed a system to prevent misdiagnosis of heart disease. Their approach is based on Neural Network (NN) and Support Vector Machine (SVM), obtaining the best results with SVM. So, according to the authors, SVM is recommended to ensure the best accuracy in diagnosis. The authors proposed to perform normalization of features in order to improve the performance of classifiers. It was decided to carry out this procedure with the data also in this thesis. In their work the authors obtained a better result with Support Vector Machine, identifying that algorithm as the best for diagnosis of heart disease.

In research paper [4], the authors implemented Neural Network with the aim to perform an efficient classification to detect heart disease. This diagnostic system appears to be an accurate tool and ensures good accuracy. With this approach the advantages of Neural Networks can be exploited, for example their use can be parallelized to improve the performance and their robustness in noisy environment. The authors proposed some fundamental steps for heart disease diagnosis: data collection, data preprocessing, outliers elimination (if present), model development, recruitment of new patients to be examined, consultation with developed model, diagnosis and evaluation by medical doctors of the diagnosis obtained with the machine learning model.

In research paper [6], the authors proposed a model with an ensemble of Neural Networks to perform heart disease prediction. The first step they proposed is to perform features removal based on their correlation coefficient. Initially, an ensemble of Neural Networks with randomly chosen parameters is considered. Only Neural Network with the best accuracy are selected. Then, to choose the best Neural Networks, entropy has been used to select the best components. In

particular, it has been used to determine the weights of the components of the Ensemble Neural Network (ENN).

In research work [5], it has been proposed a Cascaded Neural Network classifier to perform heart attack prediction. The authors obtained good results in terms of accuracy (i.e., 85%), sensitivity (i.e., 83%), specificity (i.e., 87%) and also a very short time to perform prediction. In particular, the proposed Neural Network is composed of an input unit, an hidden unit and an output unit. In their Cascade Correlation Neural Network, each hidden unit is added one by one as long as the error is not minimized. With this approach, the researchers developed a computationally efficient tool for heart attack prediction.

In summary, the proposed systems can be used as decision support system for medical staff and a method of prevention of CVDs, with the aim of ensuring accessible assistance for all people and hopefully reducing deaths around the world.
The works just presented in this section form the basis for the algorithms chosen to perform heart disease detection in this thesis.

1.3 Aim of this thesis

The purpose of this thesis is to implement and test different algorithms to perform heart disease detection. In particular, given the patients information available in public dataset, in particular in the UCI dataset, the goal is to detect if the patient is at high risk of manifesting cardiovascular diseases (CVDs) or not. Another aim is to analyze different Machine Learning algorithms and to make a comparison among their performances to find the ones that provide a better accuracy while detecting cardiovascular diseases. It is not always easy to handle large amounts of data from health examinations, and artificial intelligence could be crucial in this situation, since a lot of data coming from health care are not properly exploited for prevention.

1.4 Thesis' structure

This thesis is composed of four chapters and a final section about conclusions and future work. The second chapter presents the dataset used to perform heart disease detection and its analysis that has been performed. It also contains details about data preprocessing and all the technical details about how the code has

been implemented, including programming languages and libraries. The second part of the second chapter is dedicated to the description of how the models have been trained, with particular attention to the procedures adopted to improve the performances. The last section of this chapter describes the metrics that have been employed to quantify the models performances. The third chapter analizes in detail every algorithm that has been applied on the data, describing technical background of each machine learning model and the implementation details. The fourth chapter presents the results obtained with the different approaches, reporting metrics for each model. A critical view of the results is also proposed with a comparison with the already published works. The final chapter draws conclusions and offers proposals on future applications of machine learning for hearth diseases.

Chapter 2

Methods

In this chapter the dataset used to train the proposed models is presented. An accurate analysis has been performed on the distribution of the data. The preprocessing that has been applied on the dataset is also explained. The experimental setup is defined, including programming language, libraries and their versions. The training details of the models are specified, in particular how the hyperparameters have been chosen and which techniques have been adopted to improve performance. The metrics used to evaluate the models performances are listed and explained.

2.1 Dataset description and data preprocessing

The dataset used in the models proposed is the UCI Heart Disease Dataset [9], concerning heart disease diagnosis. This database contains medical data from four different sources: Cleveland Clinic Foundation, Hungarian Institute of Cardiology (Budapest), V.A. Medical Center (Long Beach, CA) and University Hospital (Zurich, Switzerland). The total number of attributes contained in the dataset is 76, but only 14 are used in all the currently published works since they have a clear impact on the heart disease diagnosis. The attributes are: age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak, slope, ca, thal and target. In particular, the "target" attribute refers to the presence or absence of a heart disease in the patient, diagnosed from coronary angiogram. The other 13 attributes are described in Table 2.1, in which names and descriptions are given. All of them have a numeric value. The dataset is composed of results from medical examinations.
In detail, some of them are data from routine examinations such as resting sys-

tolic blood pressure and serum cholesterol, while others are results of specific examinations such as stress electrocardiography, stress thallium scintigraphy, cine fluoroscopy and coronary angiography. In particular the target feature has been obtained performing an angiography test on the patients, categorizing them according to the level of diameter narrowing of any major vessel. The stress electrocardiography is a test that uses ultrasound imaging to show how the heart muscle is working to pump blood to the body while exercising. It is often used to detect a decrease in blood flow to the heart muscle due to narrowing in the coronary arteries. Figure 2.1 shows an electrocardiography examination during exercise of a patient[2].

Figure 2.1: Image of a patient undergoing a stress test

A thallium stress test is an imaging test performed to trace the levels of blood reaching different parts of the heart. During the procedure, a small amount of thallium, is administered into a vein on the arm. The test is performed both during rest and during exercise. The scan is helpful to detect coronary artery disease. The fluoroscopy consists in a non-invasive test to see the flow of blood through the coronary arteries in order to evaluate the presence of arterial blockages. The coronary angiography is an invasive procedure performed with the use of a catheter and x-rays to detect narrowed or blocked coronary arteries and to look for abnormalities of the patient's heart muscle or heart valves.
It is important to consider that some features have been obtained from exercise-based tests while other features have been obtained from rest-based tests. This difference is important since the examination results change a lot depending on

[2]Image credits: `https://illawarraheartcentre.com.au/exercise-stress-test-and-stress-echo/`

whether the body is performing exercise or not. In Table 2.1 it is also specified if the feature has been detected during exercise or at rest.

Attribute	**Description**	**State**
age	Age in years	-
sex	Sex (0=female, 1=male)	-
cp	Chest pain type (1=typical angina, 2=atypical angina, 3=non-anginal pain, 4=asymptomatic)	Rest
trestbps	Resting systolic blood pressure (in mmHg on admission to the hospital)	Rest
chol	Serum cholesterol in mg/dl	Rest
fbs	Fasting blood sugar $>$ 120 mg/dl (1=true, 0=false)	Rest
restecg	Resting electrocardiographic results (0=normal, 1=having ST-T wave abnormality, 2=showing probable or definite left ventricular hypertrophy by Estes' criteria)	Rest
thalach	Maximum heart rate achieved during exercise	Exercise
exang	Exercise induced angina (1=yes, 0=no)	Exercise
oldpeak	ST depression induced by exercise relative to rest	Exercise
slope	Slope of the peak exercise ST segment (1=upsloping, 2=flat, 3=downsloping)	Exercise
ca	Number of major vessels (0-3) colored by flouroscopy for coronary calcium	Rest
thal	Exercise thallium scintigraphic defect (3=normal, 6=fixed defect, 7=reversable defect)	Exercise
target	Diagnosis of heart disease (0=healthy, 1=hearth disease)	Rest

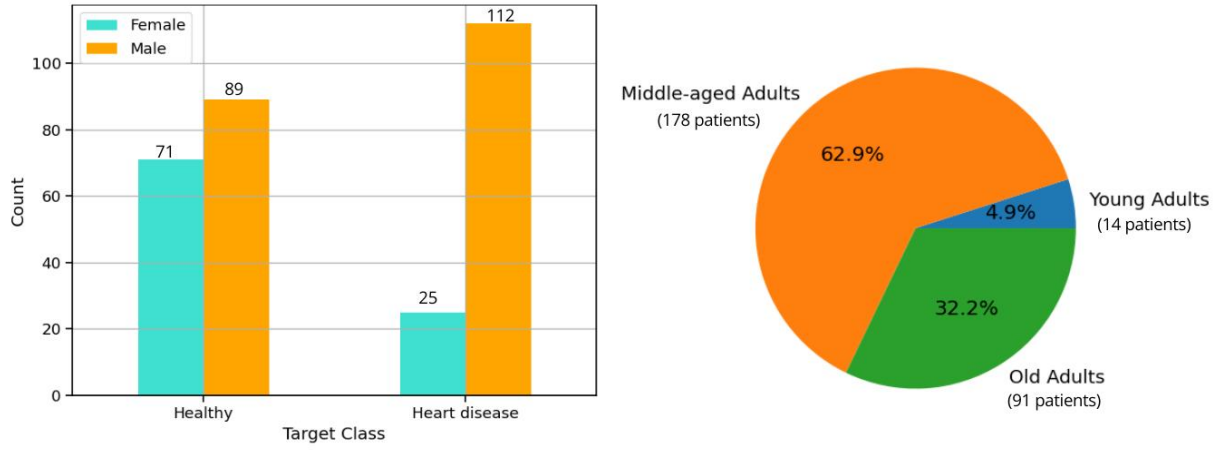
Table 2.1: Description of the attributes in the UCI Heart Disease Dataset

The number of instances from Cleveland is 303, 294 from Hungary, 123 from Switzerland, 200 from Long Beach.
The current published database presents a lot of corrupted data in datasets from Hungary, Switzerland and Long Beach. Therefore, in the proposed models only data from Cleveland have been used.

In addition, since there are some missing values, preprocessing is necessary as explained in Section 2.1.2.
As mentioned in the introductory paper of the dataset [10], there is a limitation in the data collection of this dataset. Indeed, the patients selected to perform angiography by medical doctors already presented some symptoms, so the selection was not bias free.

2.1.1 UCI Dataset Analysis

The processed database now contains data from 303 patients. It is fundamental to make an accurate analysis of each attribute to understand more details about the data before implementing machine learning algorithms. After this process, it is possible to figure out how the different features are related. In this section different graphs are shown, one for each feature, with the purpose of showing how the features are distributed in relation to the target feature. For the categorical features is reported in the graph the number of patients belonging to each value of the considered feature.



a) Target distribution for sex b) Age Distribution

Figure 2.2: Distribution of the data based on sex and age

In Figure 2.2.a, it can be observed that the class of the patients with possible heart disease is not equally distributed between the two sexes. The global impact of heart diseases has a different distribution than the one represented by this dataset, as stated in [11]. This difference may be withdrawn by expanding the dataset to a larger population. In Figure 2.2.b, the distribution of different ages

over the entire dataset is represented. In particular Young Adults class includes people between 25 and 44 years old, Middle-aged Adults class includes people between 44 and 60 years old and Old Adults class includes people over 60 years old. The age classes are computed according the World Health Organization standard [12].

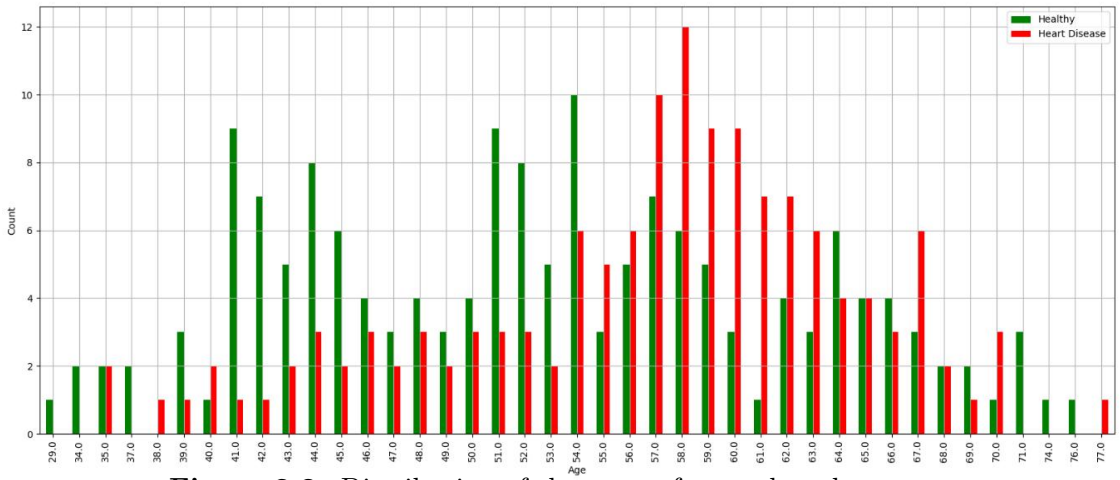


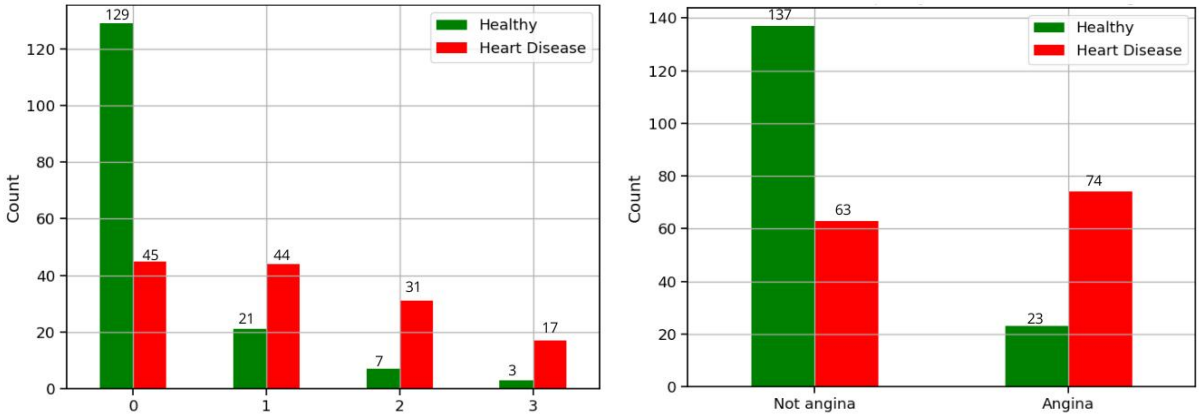
Figure 2.3: Distribution of the target feature based on age

Considering the distribution of the target feature in Figure 2.3, it can be observed that the patients in this dataset present an increase incidence of heart disease at the age of sixty.

As the number of major vessels colored by fluoroscopy increases, also the incidence of heart disease is more pronounced, this phenomenon can be explained since the calcified arteries (colored by fluoroscopy) may impair heart function. Observing the distribution of the target features in Figure 2.4.a, it can be seen how the presence of heart disease has negatively affected the results of the fluoroscopy. Considering Figure 2.4.b, it can be observed that the presence of heart disease has affected the perception of pain during exercise of the patients. In fact, the probability of being affected by heart disease is higher in patients who experienced angina.

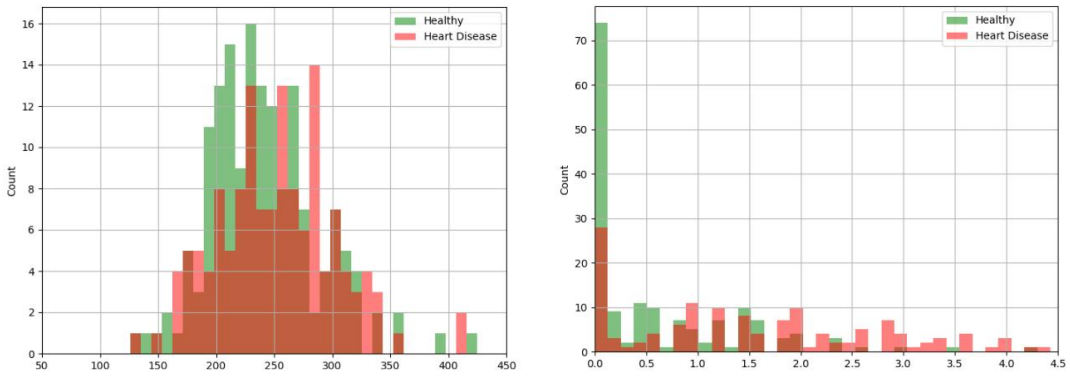
In Figure 2.5.b, it can be seen that the risk of manifesting heart diseases increase as ST depression increases. In particular, the term 'ST' refers to positions on the ECG curve and, the more the depression is marked, the higher is the risk of a heart attack.

In Figure 2.6.a it is represented the distribution of the target feature over the ECG results and it may suggest that the abnormalities correspond to an increase



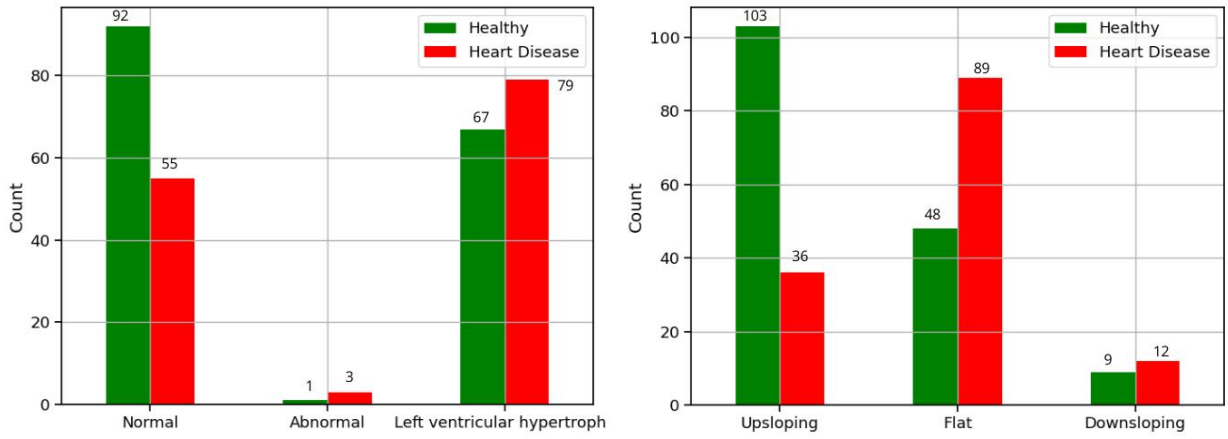
a) Number of Major Vessels colored by fluoroscopy b) Exercise induced angina

Figure 2.4: Distribution of the target feature based on results of fluoroscopy results and exercise induced angina



a) Serum Cholestoral (mg/dl) b) ST depression

Figure 2.5: Distribution of the target feature based on serum Cholestoral results and ST depression



a) Target distribution for electrocardiographic results

b) Target distribution for slope of the ST segment

Figure 2.6: Distribution of the target feature based on electrocardiographic results and slope of the ST segment

incidence of cardiovascular diseases. Figure 2.6.b represents the target distribution over the values of the slope of the ST segment. As suggested in [13], the normal ST segment of a healthy person during exercise should be upsloping. This trend corresponds to the patient data from the UCI dataset. In fact, it can be observed in Figure 2.6.b. that patients with an upsloping ST segment are mostly healthy, while patients with a flat or downsloping ST segment are mostly affected by heart disease.

In Figure 2.7 different trends of the ST segment can be observed [3].

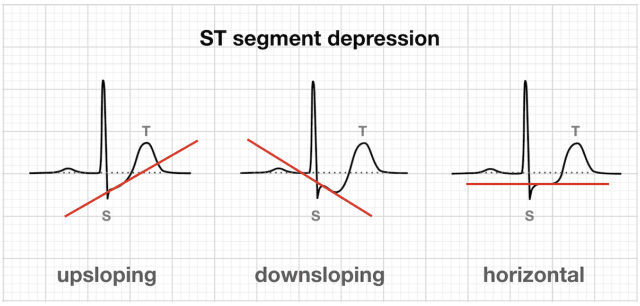
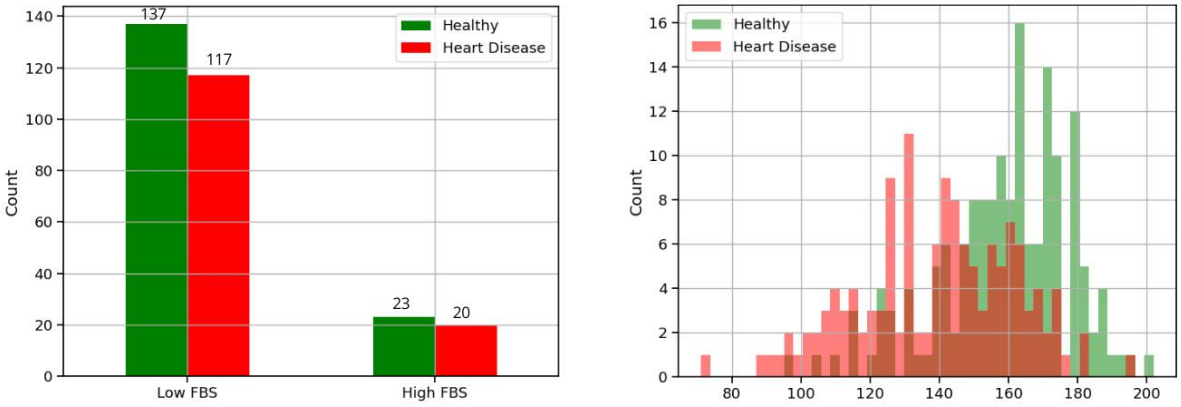


Figure 2.7: Examples of ST segments

The Figure 2.8.a represents the distribution of the target feature over the value of the fasting blood sugar exam on the patients. Generally, a fasting blood sugar

[3]Image credits: `https://litfl.com/myocardial-ischaemia-ecg-library/`



a) Target distribution for fasting blood sugar (>120 mg/dl)

b) Maximum heart rate achieved during exercise

Figure 2.8: Distribution of the target feature based on fasting blood sugar and maximum heart rate achieved during exercise

level lower than 120 mg/dL is considered normal or in some cases pre diabetic. If the value is higher than 120 mg/dL, it is possible to diagnose diabetes. Looking at the distribution developed from UCI dataset, fasting blood sugar does not seem to be impactful on the presence of heart disease. Furthermore, from the Figure 2.8.b, it can be inferred that healthy people can reach higher heart rate during exercise than people with heart disease. In Figure 2.9 the distribution of the

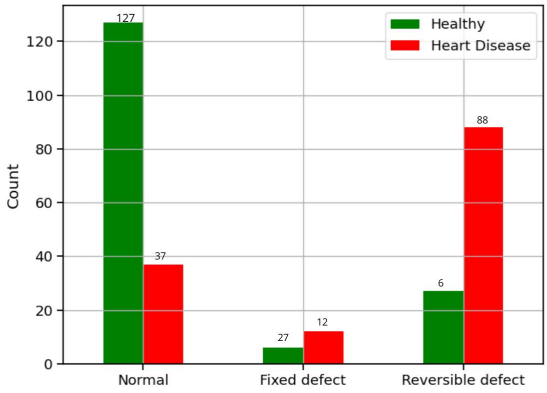
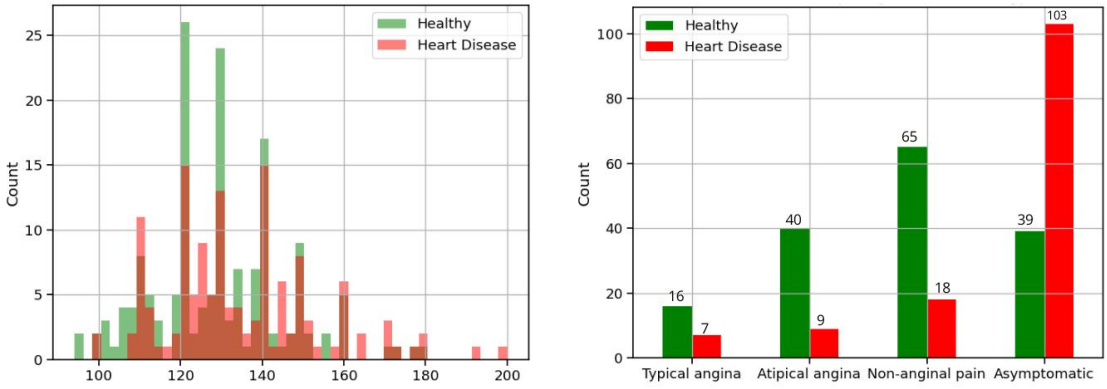


Figure 2.9: Exercise thallium scintigraphic defect distribution

thallium examination is represented, with this exam is possible to detect heart perfusion abnormalities. Result can take on three values: normal, fixed perfusion defect and reversable perfusion defect. In the patients present in UCI dataset, it seems that the result of this examination has a worse result when heart disease

is present.



a) Resting blood pressure (mmHg)

b) Heart Disease distribution for chest pain type

Figure 2.10: Distribution of the target feature based on resting blood pressure and chest pain type

Figure 2.10.b represents the distribution of the target feature over the value of the chest pain type experienced by the patients. As declared by Mayo Clinic[4], angina pectoris is characterizing for heart disease, so the distribution obtained from the UCI dataset is not an accurate description of the real symptoms of CVDs. As suggested in [14], analyzing the distribution from another prospective, it shows the importance of supporting medical decision with technology, considering that cases without angina pectoris may not always be diagnosed.

The graph shown in Figure 2.11 represents the correlation matrix with correlation coefficients between different features of the UCI dataset. The correlation matrix is helpful in data analysis to identify patterns in data and to make predictions about the trend of a feature.

The correlation coefficients quantify if two features are strongly or weakly related, with a relationship of linear correlation. Considering two different features a and b when a increases and correspondingly b increases there is a positive correlation coefficient, while when a increases and b correspondingly decreases there is a negative correlation coefficient. In general, the further a coefficient is from zero, the stronger the correlation.

The following formula explains how to calculate a generic coefficient r_{xy} that

[4] `https://www.mayoclinic.org/diseases-conditions/angina/symptoms-causes/syc-20369373#\protect\protect\leavevmode@ifvmode\kern+.2222em\relax~:text=Angina%20(an%2DJIE%2Dnuh,or%20pain%20in%20the%20chest.`

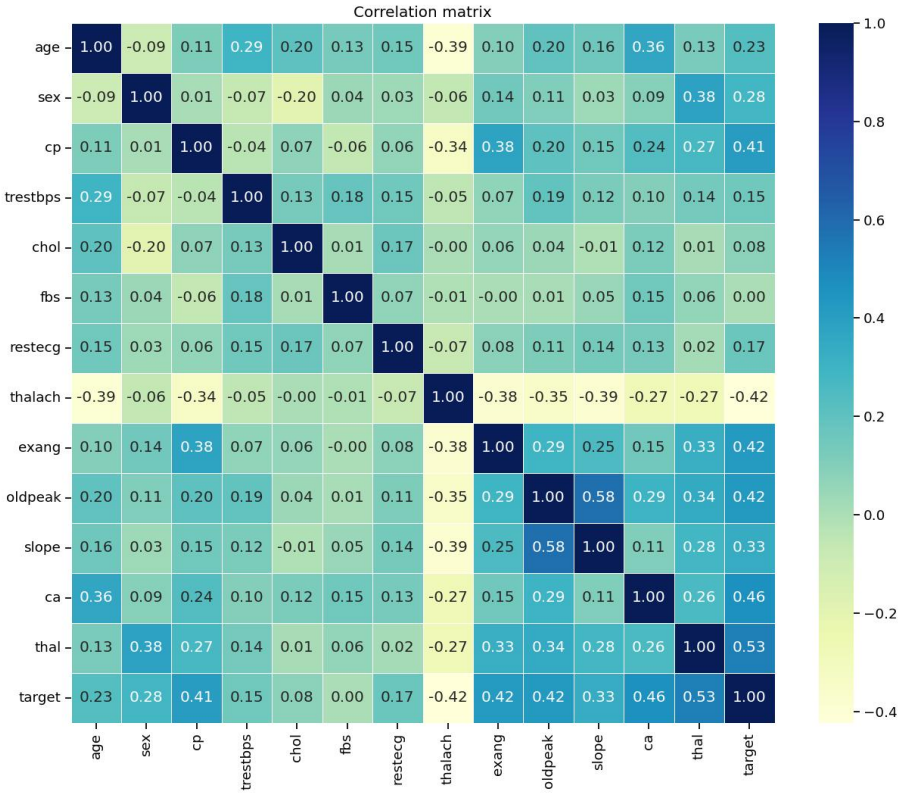
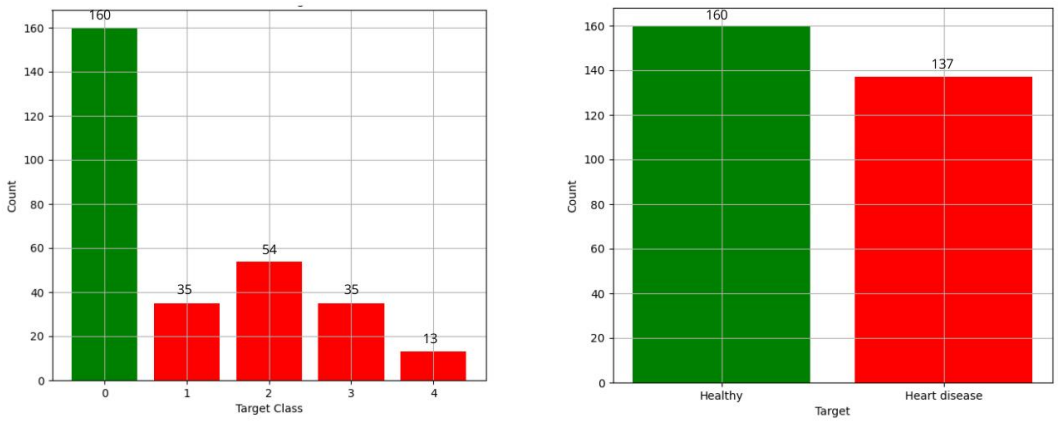


Figure 2.11: Correlation matrix of the UCI dataset features

makes up the matrix, called Pearson correlation coefficient:
$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i{}^2 - (\sum x_i)^2}\sqrt{n\sum y_i{}^2 - (\sum y_i)^2}}$
where n is the number of instances and x_i and y_i are the individual values of a feature indexed with i.
For example looking at the matrix in 2.11, the correlation coefficient between the target feature and oldpeak feature (ST depression induced by exercise) is 0.58, which means that this feature has a significant impact on diagnosis considering this dataset. Observing the matrix, it can be noted that all the coefficients in the diagonal are ones, this is because the correlation between a variable and itself is always one.

2.1.2 Data preprocessing

Data preprocessing is necessary to perform a correct analysis of the data. Since there are some missing values in the used dataset, it has been essential to remove invalid entries. At the end of this procedure there are 296 entries in total. Originally the target feature presented five different values (0, 1, 2, 3, 4), each of which indicated gravity of the heart disease. Since the four values where not equally distributed in the dataset, it has been decided to reduce this feature to two different values: presence or absence of heart disease. This choice of data processing is in line with the published papers that used the UCI Dataset. In Figure 2.12, the distribution of the target feature before and after processing is shown. After preprocessing the target feature, it has been separated from the rest of the features and used as a label to report the presence or absence of heart disease. At the end of the processing a binary classification problem must be solved. Before training the different models, the dataset is split into two parts: one for the training and one for the testing. In particular, 75% of the data has been reserved for training while the other 25% for testing.
Part of the data preprocessing is also the standardization of the dataset. It is a procedure that allows to optimize the developed algorithms by standardizing individual features to normally distributed data, that is with zero mean and unit variance. It it possible to realize this procedure with the `StandardScaler()` function. In particular, in the developed models the standardization has been implemented by developing a pipeline. The procedure, that is repeated for each feature, can be summarized as follows: $x = \frac{x-\mu}{\sigma}$, where x is the variable that indicates all the values assumed by the feature into consideration, μ is is the average of the values assumed by the feature and σ is the standard deviation.



a) Target distribution before preprocessing, each value indicates gravity of the heart disease

b) Target distribution after preprocessing

Figure 2.12: Target distribution

2.2 Experimental setup

The programming language used for the proposed approaches is Python (v. 3.10.12). The libraries used to implement the algorithms are Scikit-learn (v. 1.2.2), TensorFlow (v. 2.12.0) and Keras (v. 2.12.0). Pandas (v. 1.5.3) and Seaborn (v. 0.12.2) libraries have also been used for data visualization and data preprocessing.

Scikit-learn [5] is an open-source library for machine learning that provides tools for data analysis and developement of both supervised and unsupervised learning.

TensorFlow [6] is an open-source framework for artificial intelligence, that focuses deep learning.

Keras [7] is an open-source library that provides a powerful interface to implement neural networks and it is based on TensorFlow.

The Pandas library [8] has been used to perform data manipulation and analysis.

Seaborn library [9] has been employed to perform data visualization.

All the code has been developed and executed in Google Colaboratory[10].

[5] `https://scikit-learn.org`
[6] `https://www.tensorflow.org`
[7] `https://keras.io`
[8] `https://pandas.pydata.org`
[9] `https://seaborn.pydata.org/`
[10] `https://colab.research.google.com/`

2.3 Training

In this section the technical details of the methods applied to improve models performance are presented.

2.3.1 Hyperparameter tuning

In general, each machine learning model presents its own hyperparameters. Scikit-Learn library provides the `GridSearchCV`, which allows to perform hyperparameter tuning for a given model, finding out the values that ensure the best performance. By calling this function, the optimized hyperparameters of the classifier in question are provided. This function provides the best parameters from a list of predefined hyperparameters. The operation underlying `GridSearchCV` is to try all the combinations of the values passed to the function and to evaluate the machine learning algorithm into account for each combination using the Cross-Validation. By default, the function `GridSearchCV` uses the 5-fold cross validation. Therefore, for models developed with Scikit-Learn library, `GridSearchCV` function is exploited to identify the best hyperparameters.
To make sure the models code is reproducible, it is necessary to fix a `np.random.seed(x)` (when using the Scikit-learn library) or a `tf.keras.utils.set_random_seed(x)` (when using the TensorFlow library), where `x` is the chosen fixed number. By calling this function it is possible to obtain reproducible results for every model and to make comparisons among their performances.

2.3.2 K-fold cross-validation

K-fold cross-validation is a procedure adopted in machine learning models to evaluate each model's generalization performance. In fact, this technique is used to identify the models ability to make the correct classification on unseen data. In k-fold cross-validation the whole dataset is split into k folds. One fold is used to perform testing of the model while the other $k-1$ folds are used for the training. These steps are repeated k times and the final accuracy of the model is obtained by averaging the accuracy of each fold. In particular, in the model proposed in this thesis, it has been implemented the leave-one-out cross-validation, in which the number of folds (k) is set equal to the number of training examples. This approach is useful while working with relatively small datasets.

2.4 Metrics

Different metrics have been used to evaluate the performance of the models. In particular:

- Accuracy is defined as the ability of the classifier to predict the correct label of the samples.
  $Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$

- Precision is defined as the ratio between two classes of samples labeled by the classifier: true positive samples and the total of positive-labeled samples (this latter includes both true and false positive).
  $Precision = \frac{TP}{TP+FP}$

- Recall is defined as the ratio between two classes of samples labeled by the classifier: true positive samples and the total of samples that should be classified as positive.
  $Recall = \frac{TP}{TP+FN}$

Another used metric is the time taken by the algorithm for classification, expressed in seconds.
To assess the performance of a machine learning model the confusion matrix is often used, as it has been done in this thesis. A confusion matrix is a graphical representation that shows the number of true positive, false positive, true negative and false negative classifications of the model under consideration. In fact, the above mentioned sets are computed comparing the actual labels of the examples with the predicted classification of the model. An example of confusion matrix is shown in Figure 2.13.
In the considered model true positive samples include healthy patients that have been labeled as healthy by the algorithm, true negatives samples include ill patients that have been labeled as ill by the algorithm, false positive samples include ill patients that have been labeled as healthy by the algorithm and false negatives samples include healthy patients that have been labeled as ill by the algorithm.

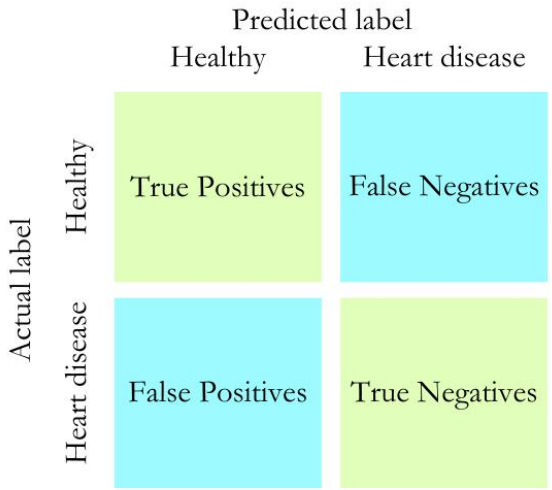


Figure 2.13: Example of confusion matrix

Chapter 3

Models

In this chapter the machine learning models chosen to perform heart disease detection are presented, starting from their technical background to their implementation in Python. All the proposed algorithms are based on supervised learning.

3.1 Neural Network

Neural Networks (NNs) are deep learning computational models. NNs are composed of an input layer, one or more hidden layers, and an output layer. Every layer is composed of different nodes, each node receives an weighted input from the previous layer and computes an output for the next layer applying a nonlinear function.
The model selected to perform heart disease detection in this thesis is the Feed-forward Network, so the connections among nodes is only in one direction and the resulting representation in a directed acyclic graph, as shown in Figure 3.1.

3.1.1 Algorithm implementation

To implement the Neural Network in Python, the Tensorflow and Keras libraries[11] have been used.
Neural Networks model requires different layers, in particular to perform heart disease detection three different Dense layers have been used, as suggested in [3]. These layers have different activation functions, the first two use the rectified linear unit (ReLU) function, while the last layers uses a sigmoid function. Also,

[11] `https://colab.research.google.com/drive/15OCRMSt8-r7P2WB-w3nIosC_-pEuuz3R?usp=sharing`

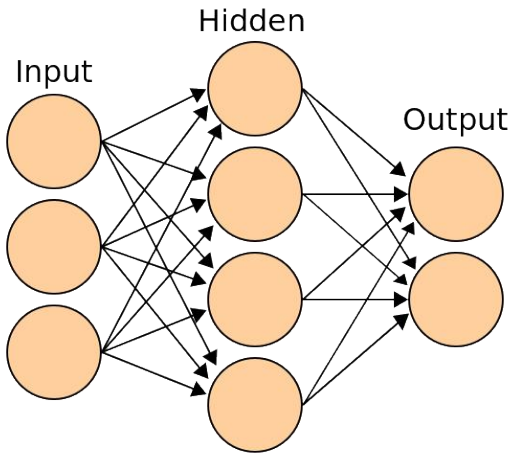


Figure 3.1: Example of Neural Network

each layer has a different number of internal nodes: the first one has 13 nodes, the second one has 6 nodes and the last one has 1 node, since a binary classification has been performed. A representation of the architecture of the proposed network is shown in the Figure 3.2.

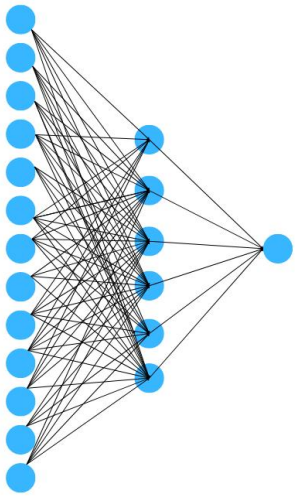


Figure 3.2: Architecture of the proposed network

The number of epochs that allow to obtain the highest accuracy is 150.

3.2 Support Vector Machine

Support Vector Machine (SVM) is a linear model to perform classification. It has some important characteristics: SVMs constitute a maximum margin separator among points and by separating the points they create a linear hyperplane. SVMs are a non-parametric model so they have the ability to represent complex functions. With the kernel trick SVMs can embed the entry data into a higher-dimensional space. The kernel trick takes advantages of the kernel functions to manipulate the data. The most commonly used functions are linear, radial basis, polynomial and sigmoid.

3.2.1 Algorithm implementation

SVM algorithm has been implemented by using the Scikit-learn library.[12]
Different kernel functions have been applied to the SVM algorithm: linear function, radial basis function, polynomial function and sigmoid function. The formal definition of these kernel functions are[13]:

- Linear: $< x, x' >$
- Radial basis: $exp(-\gamma||x - x'||^2)$
- Polynomial: $(\gamma < x, x' > +r)^d$
- Sigmoid: $tanh(\gamma < x, x' > +r)$

where d is fixed by parameter `degree`, r is fixed by parameter `coef0` and γ is specified by parameter `gamma`. The parameters chosen to train the model were identified with the `GridSearchCV` function, as explained in Section 2.3. The best parameters returned by the `GridSearchCV` function are different for every kernel function. In particular:

- Linear:`C`=0.1, `gamma`=scale.
- Radial basis: `C`=100, `gamma`=scale.
- Polynomial: `C`=100, `gamma`=scale.
- Sigmoid: `C`=10, `gamma`=scale.

[12] `https://colab.research.google.com/drive/1i8gO6Sc-lql_wZ76tSWxRsKxvbYm_El-?usp=drive_link`
[13] `https://scikit-learn.org/stable/modules/svm.html#kernel-functions`

The parameter `C` is a regularization parameter that controls the ability of generalization of the classifier into account and parameter `gamma` defines the impact of a singe example on the classification.

3.3 Naive Bayes Classifier

Naive Bayes is a Machine Learning classifier built on Bayes' theorem. Bayes' theorem is based on the definition of conditional probability and it is expressed with the following equation:

$$P(a|b) = \frac{P(b|a)P(a)}{P(b)}$$

where $P(a)$ is the probability of event a occurring, $P(b)$ is the probability of event b occurring, $P(b|a)$ is the probability of event b occurring given event a has occurred, $P(a|b)$ is the probability of event a occurring given event b has occurred. This important equation is used to determine the cause given effect of some unknown cause. Bayes' rule can be written as

$$P(cause|effect) = \frac{P(effect|cause)P(cause)}{P(effect)}$$

where $P(effect|cause)$ quantifies the causal relation and $P(cause|effect)$ quantifies the diagnostic relation. Since the classification is binary, the following function is applied to obtain the output class from the algorithm

$$y = argmax_y P(y) \prod_{i=1}^{n} P(x_i|y)$$

where $x_1, ..., x_n$ are the values of the features and y represent the class variable.

3.3.1 Algorithm implementation

Naive Bayes algorithm has been implemented using Scikit-learn library[14]. The parameters chosen to train the model were identified with the `GridSearchCV` function, as explained in Section 2.3. The best parameter returned by the `GridSearchCV` function is `var_smoothing`=1e-9, where `var_smoothing` is a parameter with the purpose of calculation stability.

[14] `https://colab.research.google.com/drive/1g5B7r721Coh5NwNVUFtBeFwF0gxoKgmK?usp=drive_link`

3.4 Stochastic Gradient Descent

Stochastic Gradient Descent (SGD) is a model used to perform fitting of linear classifiers, included linear Support Vector Machines. SGD is sensitive to feature scaling, procedure explained in Section 2.3.1.

3.4.1 Algorithm implementation

Stochastic Gradient Descent algorithm has been implemented using Tensorflow[15]. The parameters chosen to train the model were identified with the `GridSearchCV` function, as explained in Section 2.3. The best parameters returned by the `GridSearchCV` function are `loss`=squared_hinge, `penalty`=l1, and `max_iter`=100. The parameter `loss` describes the loss function to be applied (hinge means a linear SVM), parameter `penalty` defines the regularization term and parameter `max_iter` defines the number of epochs.

3.5 Decision Tree Classifier

Decision Tree is a Machine Learning classifier based on a tree-like structure. This type of classifier makes a decision about what value to assign to the output class. In particular, the algorithm starts from the root node and follows the proper branch to reach the next decision node until it reaches a leaf node. A Decision Tree is composed of:

- Internal nodes: constitute a test of the value of a feature.
- Branches: represent connections of features which lead to the leaf nodes.
- Leaf nodes: specify the output class returned by the algorithm.

3.5.1 Algorithm implementation

Decision tree algorithm has been implemented using Scikit-learn library[16].
The parameters chosen to train the model were identified with the `GridSearchCV` function, as explained in Section 2.3. The best parameters returned by the

[15] `https://colab.research.google.com/drive/1fOreDByk0F4KPmbC3j8pXLJxff7MXV7Q?usp=sharing`

[16] `https://colab.research.google.com/drive/1gaAaa6I58JRYHKw8EzUg3tfZ_z5QUAXo?usp=drive_link`

`GridSearchCV` function are `criterion`=log_loss, `max_features`=log2. The parameter `criterion` is a function which measures the quality of a split and the parameter `max_features` is the number of features to consider while searching for the best split.

3.6 Logistic Regression

Logistic Regression (LR) is a classification model that uses a logistic function (sigmoid function) to determine the probability of a variable to belong to a certain class or not. The dependent variable is binary in heart disease detection (presence or absence of heart disease). In LR, the dependent variable is the target variable that the model is trying to predict.

3.6.1 Algorithm implementation

Logistic regression algorithm has been implemented using Scikit-learn library[17]. The parameters chosen to train the model were identified with the `GridSearchCV` function, as explained in Section 2.3. The best parameters returned by the `GridSearchCV` function are `penalty`=l2, `max_iter`=50 and `C`=100. The parameter `penalty` defines the regularization term, parameter `max_iter` defines the number of epochs and the parameter `C` is a regularization parameter that controls the ability to generalize of the classifier into account.

[17] `https://colab.research.google.com/drive/1JC96RSdXvLvrkv1RWIw2kc4O-foGnape?usp=sharing`

Chapter 4

Results

In this chapter the performances of all the implemented algorithms are presented, including accuracy, precision, recall, time and the confusion matrixes of each model. Section 4.2 presents a critical discussion of the results, including a comparison with the already published works.

4.1 Performances

In Table 4.1 the metrics of the models on the training phase are reported with the use of the K-fold cross validation, explained in Section 2.3. The results are obtained by calling the function `cross_val_score`, which allows to see the value of accuracy of each fold. The values reported in the table are the average of the accuracy of all folds, as well as the standard deviation is computed considering all folds. In this way the performance for the entire dataset is provided.
`cross_val_score` is a function which allows to run cross validation on a dataset to test the ability of the models. `cross_val_score` has been used to train and test the model on multiple folds with the aim to ensure that the model generalises reasonably well across the whole dataset and not just for a single portion, as happens when the dataset is split into training and testing sets.

Algorithm	Accuracy	SD	Time
Logistic Regression	0.82	0.38	1.851 s
Naive Bayes	0.84	0.37	0.687 s
Decision Tree	0.70	0.46	0.828 s
Stochastic Gradient Descent	0.69	0.46	1.510 s
SVM (Linear Function Kernel)	0.83	0.38	1.925 s
SVM (Radial Basis Function Kernel)	0.76	0.42	3.971 s
SVM (Polynomial Function Kernel)	0.76	0.42	3.176 s
SVM (Sigmoid Function Kernel)	0.82	0.38	1.550 s
Neural Network	0.87	0.34	2.687 s

Table 4.1: Results with K-fold cross validation, where SD is the Standard Deviation of accuracy, Time is the time (in seconds) it took the model to complete the training

Table 4.2 reports the metrics of the final models on the testing phase, to verify the performances on unseen data.

Algorithm	Accuracy	Precision	Recall	Time
Logistic Regression	87%	89%	79%	0.027 s
Naive Bayes	**90.6%**	**96.4%**	81.8%	**0.003 s**
Decision Tree	77%	70.2%	78.7%	**0.003** s
Stochastic Gradient Descent	88%	93%	78.7%	0.008 s
SVM (Linear Function Kernel)	89.3%	90.3%	**84.8%**	0.007 s
SVM (Radial Basis Function Kernel)	77%	70%	84.8%	0.011 s
SVM (Polynomial Function Kernel)	82.6%	86%	72.7%	0.014 s
SVM (Sigmoid Function Kernel)	84%	88%	72.7%	0.006 s
Neural Network	88%	87.5%	**84.8%**	0.057 s

Table 4.2: Results of the testing set

4.1.1 Confusion Matrices

The confusion matrices achieved per each model are shown in this subsection. The label 0 stands for healthy and the label 1 stands for possible presence of heart disease. They can be useful to have a clear visual representation of the

various models performances. It is possible to compute the values of accuracy, precision and recall by consulting confusion matrices, since they contain the values explained in Section 2.4.

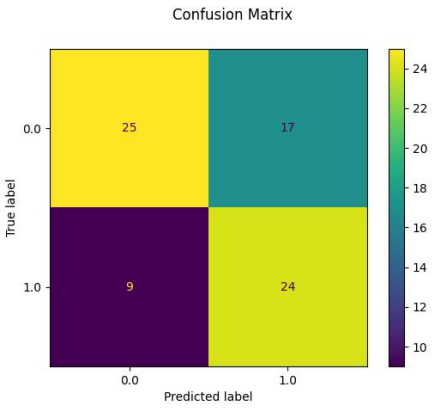


Figure 4.1: Decision tree

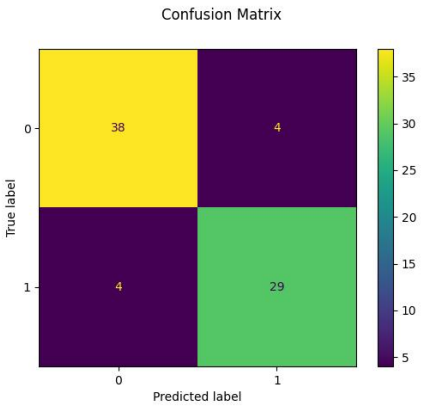


Figure 4.2: Logistic regression

For example in Figure 4.1, when using Decision tree algorithm, it can be observed that 25 patients have been predicted as healthy and also the actual label is negative to heart disease (true negative). 24 patients have been predicted with presence of heart disease and also the actual label is positive to heart disease (true positive).

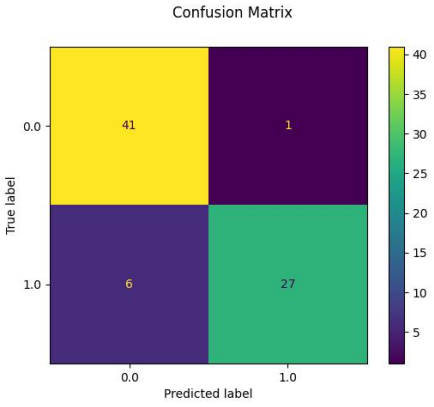


Figure 4.3: Naive Bayes

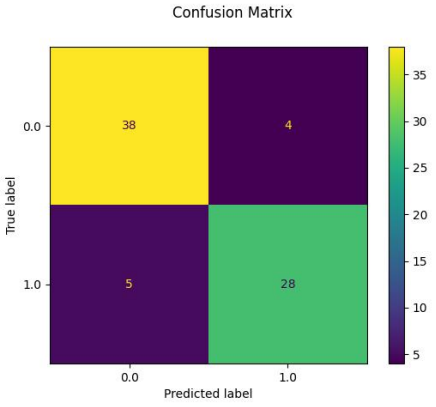


Figure 4.4: Neural network

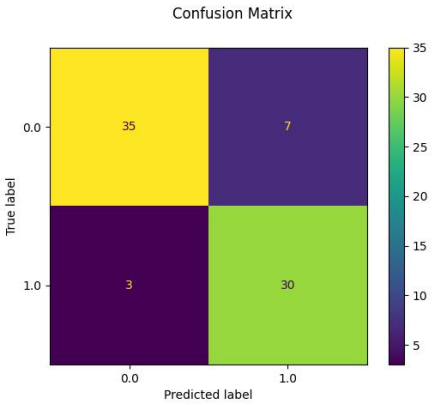


Figure 4.5: Stochastic gradient descent

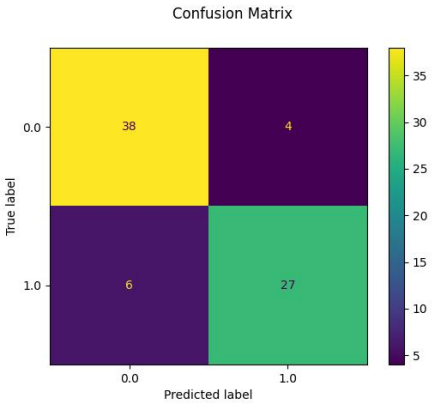


Figure 4.6: SVM (linear kernel)

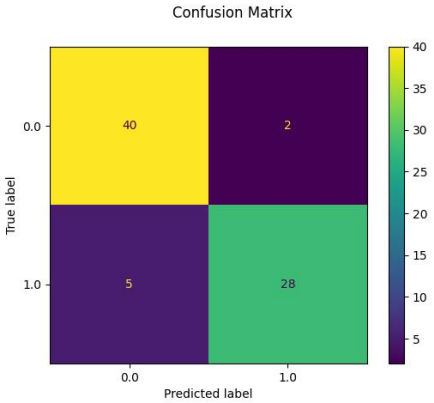


Figure 4.7: SVM (polynomial kernel)

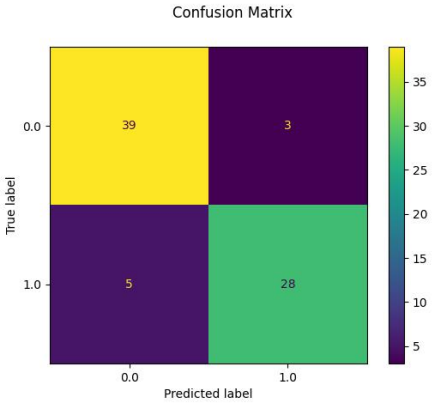


Figure 4.8: SVM (sigmoid kernel)

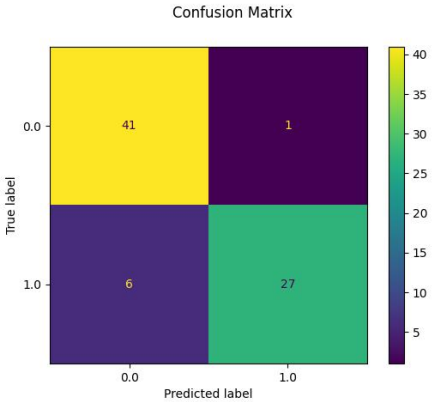


Figure 4.9: SVM (rbf kernel)

4.2 Discussion

From the results in Table 4.2, a comparison can be made among the performances from the different models. Considering accuracy, the model that gave the best performance is Naive Bayes, that has shown 90.6% accuracy.
Model that has demonstrated highest precision is Naive Bayes with 96.4%. Neural Network and SVM (Linear Function Kernel) are the models that have proven to have the highest recall with a value of 84.8%.
Lastly, the models that performed the heart disease detection on the dataset in the shortest time are Naive Bayes and Decision Tree with a time of 0.003 seconds.

Lastly, by making a comparison with the performances of the same models already published by researchers, it can be observed that the results obtained with the models developed in this thesis are aligned with the ones presented in Table 1.1. The Naive Bayes model proposed in this thesis achieved a better result in terms of accuracy compared to the one proposed in work [8].
Considering that each model can have different implementations and hyperparameters, the ones chosen in this thesis seem to guarantee a good overall performance in heart disease detection.
However, the proposed models have limitations mainly represented by the used dataset. In fact, due to the relatively small population of patients who have consented to collect their data for research purposes, their data probably may not represent an accurate distribution of the trend of the features in the entire world population. Also, as indicated in the introductory paper of the UCI dataset [10], there are numerous biases coming from the choice of patients to be recruited for the examinations.

Conclusions and future work

In conclusion it has been demonstrated that the proposed algorithms have a good performance in detecting heart disease and that the obtained results are aligned with those of the published papers. This result can pave the way for other applications in healthcare. A limitation that has been identified in the developed models is the relatively small size of the UCI dataset that does not fully reflect the real health data of the world population. For instances, in terms of the value of chest pain perceived by the patients and the distribution of the target feature over the two sexes. In fact, these two features belonging to the examined dataset show a incoherent trend compared to the already verified medical knowledge.

One possible future implementation of the proposed approaches is to develop an application for both patients and medical doctors. In this way, the patients could enter information that do not need a medical intervention and consult independently the results of their health examinations. The doctors can continuously monitor their patients and identify those with high risk of manifesting CVDs without manually comparing all the features.

Another future development is to integrate these algorithms into with bracelets or watches equipped with sensors. These devices can detect abnormal heart rate values and notify the patient/doctor when the algorithm considers that there is a high risk of heart disease arising. An important consideration is that the symptoms of heart diseases are not always evident while performing a medical exam, so the use of sensors can guarantee control even when the medical staff is not present, and a wider and more detailed overview of the patient's situation.

It is essential to lower the cost of health examinations to make them more accessible in order to collect more data and make the results more reliable. Moreover, by making the examinations more accessible, they would be performed more regularly, allowing the best use of the proposed medical systems that could properly alert patients to the need for a medical consultation. Lastly, it could be useful to create a shared dataset with anonymous data, then used to train machine learn-

ing models with the aim to increase the accuracy of the proposed models and leverage the data that are constantly produced by the medical industry but are not sufficiently exploited.
The proposed algorithms are designed to support doctors and patients and are not a replacement for doctors. Prevention is achieved by cooperation between artificial intelligence and healthcare professionals.

Bibliography

Chapter 1

[1] M. Kavitha et al. "Heart Disease Prediction using Hybrid machine Learning Model". In: *2021 6th International Conference on Inventive Computation Technologies (ICICT)*. 2021, pp. 1329–1333. DOI: 10.1109/ICICT50816.2021.9358597 (cit. on p. 2).

[2] Boyang Peter Dun, Eric T Wang, and Sagnik Majumder. "Heart Disease Diagnosis on Medical Data Using Ensemble Learning". In: 2017. URL: https://api.semanticscholar.org/CorpusID:28684079 (cit. on p. 2).

[3] Ebenezer Obaloluwa Olaniyi, Oyebade Kayode Oyedotun, and Khashman Adnan. "Heart Diseases Diagnosis Using Neural Networks Arbitration". In: vol. 7 12. 2015, pp. 75–82. DOI: 10.5815/ijisa.2015.12.08 (cit. on pp. 2, 3, 23).

[4] Azadeh Sayad and Pratap P. Halkarnikar. "DIAGNOSIS OF HEART DISEASE USING NEURAL NETWORK APPROACH". In: 2014. URL: https://api.semanticscholar.org/CorpusID:212551571 (cit. on pp. 2, 3).

[5] R. Chitra and Dr.V. Seenivasagam. "Heart Attack Prediction System using Cascaded Neural Network". In: 2013. URL: http://bonfring.org/conference/papers/xavier_icamtcs2013/tt3_tcs_0311.pdf (cit. on pp. 2, 4).

[6] S. Silvia Priscila and M. Hemalatha. "Improving the Performance of Entropy Ensembles of Neural Networks (EENNS) on Classification of Heart Disease Prediction". In: 2017. URL: https://api.semanticscholar.org/CorpusID:201110435 (cit. on pp. 2, 3).

[7] Eka Miranda et al. "Intelligent Computational Model for Early Heart Disease Prediction using Logistic Regression and Stochastic Gradient Descent (A Preliminary Study)". In: *2021 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI)*. Vol. 1. 2021, pp. 11–16. DOI: 10.1109/ICCSAI53272.2021.9609724 (cit. on pp. 2, 3).

[8] Ramesh Tr et al. "PREDICTIVE ANALYSIS OF HEART DISEASES WITH MACHINE LEARNING APPROACHES". In: *Malaysian Journal of Computer Science* 2022 (Apr. 2022), pp. 132–148. DOI: 10.22452/mjcs.sp2022no1.10 (cit. on pp. 2, 33).

Chapter 2

[9] Andras Janosi et al. *Heart Disease*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C52P4X. 1988 (cit. on p. 7).

[11] Zujie Gao et al. "Gender differences in cardiovascular disease". In: *Medicine in Novel Technology and Devices* 4 (2019), p. 100025. ISSN: 2590-0935. DOI: https://doi.org/10.1016/j.medntd.2019.100025. URL: https://www.sciencedirect.com/science/article/pii/S2590093519300256 (cit. on p. 10).

[13] Jonathan Hill and Adam Timmis. "Exercise tolerance testing". In: *"Exercise tolerance testing.* 324 (2002). DOI: 10.1136/bmj.324.7345.1084 (cit. on p. 13).

[14] Khadijah Alfadli and Alaa Almagrabi. "Feature-Limited Prediction on the UCI Heart Disease Dataset". In: *Computers, Materials & Continua* 74 (Jan. 2023), pp. 5871–5883. DOI: 10.32604/cmc.2023.033603 (cit. on p. 15).

Chapter 3

[3] Ebenezer Obaloluwa Olaniyi, Oyebade Kayode Oyedotun, and Khashman Adnan. "Heart Diseases Diagnosis Using Neural Networks Arbitration". In: vol. 7 12. 2015, pp. 75–82. DOI: 10.5815/ijisa.2015.12.08 (cit. on pp. 2, 3, 23).

[12] Akhmet Dyussenbayev. "The Main Periods of Human Life". In: *Global Journal of Human-Social Science* 17.A7 (May 2017), pp. 33–36. URL: https://socialscienceresearch.org/index.php/GJHSS/article/view/2393 (cit. on p. 11).

Chapter 4

[8] Ramesh Tr et al. "PREDICTIVE ANALYSIS OF HEART DISEASES WITH MACHINE LEARNING APPROACHES". In: *Malaysian Journal of Computer Science* 2022 (Apr. 2022), pp. 132–148. DOI: 10.22452/mjcs.sp2022no1.10 (cit. on pp. 2, 33).

Acknowledgments

I would like to express my gratitude to my advisor, Prof.Gloria Beraldo, for giving me the opportunity to work on a topic that I'm passionate about and for her valuable advice.

I am deeply thankful to my parents for all the sacrifices they make for me.

A special thanks to Alice for being my best friend and for always encouraging me to give my best.

Lastly, i would also like to thank my friends for their continuous support.