Predicting Heart Disease using Machine Learning Techniques: A Comparative Analysis

1st Josip Ivancevic 2nd Nikolas Recke
Departamento de Electrónica, Telecomunicações e Informática
Universidade de Aveiro
Aveiro, Portugal
josip.ivancevic@fer.hr
nikolas.recke@rwth-aachen.de

***Abstract*—In this paper, we present a comparative analysis of four different machine learning models: Linear Regression (including Ridge and Lasso regression), Logistic Regression, Neural Networks, and Support Vector Machines (SVM) to predict heart disease using a dataset combining a total of 920 instances and 75 attributes. We discuss the data preprocessing steps, model selection, hyperparameter tuning, and performance evaluation to provide a comprehensive understanding of each model's efficacy in predicting heart disease. After training and testing the different models, we compare our approach to the approaches of other authors who worked with this dataset.**

***Index Terms*—ML, machine learning algorithms, heart disease prediction**

I. Introduction

Cardiovascular diseases, including heart disease, are among the leading causes of morbidity and mortality worldwide. Early detection and prevention of heart disease are crucial to reducing the burden on healthcare systems and improving patient outcomes [1]. Machine learning techniques have shown great potential in the analysis of complex datasets and the prediction of various medical conditions, including heart disease. By using machine learning models, researchers and healthcare professionals can identify patterns and relationships in the data that may not be apparent through traditional statistical methods. [2]

A. Background and motivation

The application of machine learning in the medical domain has been growing rapidly in recent years, driven by advancements in computational power and the availability of vast amounts of data [2]. Machine learning models have been successful in diagnosing and predicting a wide range of medical conditions, enabling more accurate and efficient patient care [3]. The motivation behind this study is to compare the performance of four machine learning algorithms in predicting heart disease, providing insights into their strengths and limitations and informing the choice of appropriate models for future research and clinical applications.

B. Problem statement and objectives

The primary objective of this study is to develop and compare the performance of four machine learning models - Linear Regression, Logistic Regression, Neural Networks, and SVM in predicting heart disease using a dataset from the database of the UCI Machine Learning Repository [4]. The dataset contains a mix of categorical, integer, and real attributes, and includes instances with missing values, posing challenges for data preprocessing and model training. Our specific goals are as follows:

1) Preprocess the dataset by handling missing values and normalizing the data as well as pre-processing the categorical and integer attributes
2) Train and fine-tune each of the four models on the preprocessed data.
3) Evaluate the performance of the models using appropriate evaluation metrics, such as accuracy, precision, recall, F1-score, and AUC-ROC.
4) Analyze and discuss the results, highlighting the strengths and limitations of each model, and their suitability for heart disease prediction.
5) Compare results with the results of authors and find potential ways to improve the performance.

C. Paper organization

This paper is organized into the following sections. Section II provides an overview of the dataset used in the study, explaining attributes, their distribution, and challenges concerning the data. In Section III, we explain the preprocessing steps, including handling missing values and normalization. In Section IV, we introduce the machine learning models used in the study, explaining their underlying concepts and mechanisms. We then continue to explain the model selection process and hyperparameter tuning in Section V, where we discuss cross-validation and grid search. Section VI presents the evaluation and comparison metrics that we employ for the models. In Section VII, we provide the results of the models in predicting heart disease, including model hyperparameters, confusion matrices, and a comparison between all models. Section VIII offers a comparison of our results with those of other authors, assessing the relative performance and effectiveness of our models. Finally, in Section IX, we discuss the key findings, as well as the strengths and limitations of the models considering their abilities in the context of heart disease prediction. We will outline limitations and recommendations for future research.

II. Dataset Description

A. Data Source and Attributes

The heart disease datasets used in this study were collected from four sources:

- Hungarian Institute of Cardiology, Budapest: Dr. Andras Janosi
- University Hospital, Zurich, Switzerland: Dr. William Steinbrunn
- University Hospital, Basel, Switzerland: Dr. Matthias Pfisterer
- V.A. Medical Center, Long Beach, and Cleveland Clinic Foundation: Dr. Robert Detrano

The four datasets from Cleveland, Hungary, VA (V.A. Medical Center in California) and Switzerland consist in total 920 instances. The Cleveland dataset consists of 303 instances. The Hungarian, Swiss and Californian dataset consists of 294, 123 and 200 instances respectively. Although the datasets contain 75 attributes, only a subset of these attributes have been used in former publications, namely a subset of 14 attributes that will be explained in Table I and used within throughout this analysis. There, we will also provide the range of the values and explain whether the variable is continous, integer or categorical. These attributes consist of various factors related to heart disease, such as age, sex, chest pain type, blood pressure, cholesterol level, and electrocardiographic results, among others.

TABLE I
Description of variables in the heart disease dataset

Variable	Meaning	Range
age	Age in years	Continuous
ca	Major vessels	0-3
chol	Cholesterol	Continuous
cp	Chest pain type	1: TA 2: AA 3: NAP 4: Asym
diag	Angiographic status	0: $< 50\%$ 1: $> 50\%$
exang	Exercise angina	0 = No, 1 = Yes
fbs	Fasting blood sugar	0 = False, 1 = True
oldpeak	ST depression	Continuous
restecg	Resting ECG results	0: N 1: ST-T 2: LVH
sex	Sex	0 = Female, 1 = Male
slope	ST segment slope	1: Up 2: Flat 3: Down
thal	Thalium test	3: N 6: FD 7: RD
thalach	Max heart rate	Continuous
trestbps	Resting BP	Continuous

The following table II provides descriptive statistics of variables in a heart disease dataset. It includes the count, mean, and standard deviation for each variable. The count displays the total occurence among all datasets.

TABLE II
Descriptive statistics of variables in the heart disease dataset

Variable	Count	Mean	Std. Dev.
age	920	53.51	9.42
sex	920	0.79	0.41
cp	920	3.25	0.93
trestbps	861	132.13	19.07
chol	890	199.13	110.78
fbs	830	0.17	0.37
restecg	918	0.60	0.81
thalach	865	137.55	25.93
exang	865	0.39	0.49
oldpeak	858	0.88	1.09
slope	611	1.77	0.62
ca	309	0.68	0.94
thal	434	5.09	1.92
diagnosis	920	0.99	1.14

The mean age of patients in the dataset is 53.51 years, with a standard deviation of 9.42. The dataset is predominantly composed of male patients (79%), with a mean of 0.79 and a standard deviation of 0.41. The features with the most missing data are ”ca” and ”thal,” with 309 and 434 counts respectively. These features have more than 50% missing data, which can make it difficult to draw conclusions based on them. There are 10 features with more than 800 counts, which provides a good amount of data to analyze these variables for prediction capabilities. The variables with the highest standard deviation are ”thal” and ”oldpeak,” with values of 1.92 and 1.09 respectively. This indicates that these features have a wider range of values compared to the other variables in the dataset. This dataset provides valuable information on various features that can contribute to heart disease diagnosis, with a few features having more missing data than others. In the following section, we will continue to visualize the distribution of the data.

B. Data visualization

Figure 1 shows a histogram of features for the heart disease dataset. The x-axis represents the range of values for each feature, while the y-axis represents the frequency of occurrence. It is observed that some features are more close to being normally distributed (e.g. age), while others show a skewed distribution (e.g. trestbps).

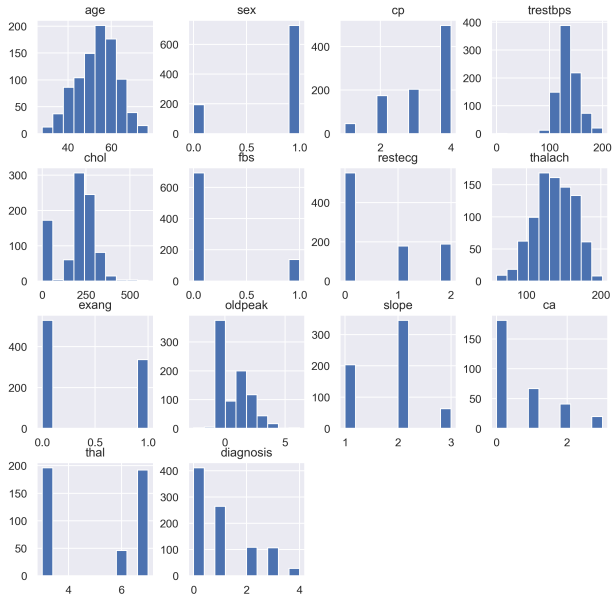

Fig. 1. Histogram of features for Training dataset.

In Figure 1 we also observe the imbalance of the diagnosis, showing that around 150 . Values for the presence of a heart disease range from 1 to 4, while absence is indicated by 0. We transformed the data to become binary, to receive a more balanced dataset. Figure 2 displays the distribution of diagnosis in the heart disease dataset. The x-axis represents the type of diagnosis (0: absence, 1: presence), while the y-axis represents the number of occurrences.

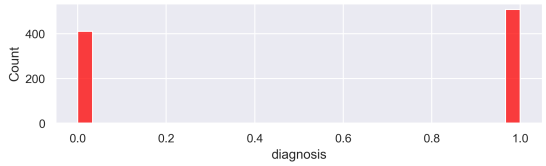

Fig. 2. Distribution of diagnosis for training dataset.

C. Associated Tasks and Challenges

Working with real-world datasets, especially in the medical domain, presents challenges that need to be addressed to ensure the effectiveness and reliability of the models developed. Some of the key tasks and challenges associated with this project are e.g. the handling of missing data: The dataset contains missing values (e.g. the attribute ca, that has more than 50% missing data), which can impact the performance of the machine learning models. Appropriate techniques, such as imputation or removal of instances with missing values, need to be employed to ensure the integrity of the dataset used for model training and evaluation. [5]. These steps will be discussed in the following section.

III. Data Preprocessing

The data preprocessing is an essential step before training any machine learning model, as it helps to clean, format, and organize the data, making it suitable for analysis [6]. In this section, we present the preprocessing steps taken, which include handling missing values, and normalization, and train-test split. We will discuss each step shortly to provide an understanding of the data preprocessing stage.

A. Feature selection

The dataset contains 14 attributes, including continuous, discrete, and binary variables. We have divided these features into two groups: continuous and discrete factors. Continuous factors include age, cholesterol levels, oldpeak, maximum heart rate, and resting blood pressure. These variables represent critical aspects of a patient's medical history and are directly related to heart disease risk. Discrete factors consist of chest pain type, exercise-induced angina, fasting blood sugar, resting electrocardiographic results, sex, slope of the peak exercise ST segment, number of major vessels, and thalassemia. These variables provide additional insights into a patient's condition and can help improve the model's ability to predict heart disease. According to [4] all contributions worked with the preprocessed dataset, based upon the 13 features that have already been selected. It is important to mention, that the reduction from 75 to 13 features is already a performed feature selection. To understand the influence of the features better, we continue to analyse the correlation coefficients of the features with the diagnosis. The correlation between different features in the heart disease dataset is represented in Figure 3.

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	diagnosis
age	1.000	0.057	0.166	0.244	-0.086	0.234	0.213	-0.366	0.202	0.258	0.155	0.370	0.137	0.340
sex	0.057	1.000	0.170	0.001	-0.197	0.089	-0.016	-0.179	0.181	0.104	0.125	0.094	0.374	0.259
cp	0.166	0.170	1.000	0.025	-0.133	0.040	0.031	-0.349	0.419	0.244	0.203	0.215	0.313	0.398
trestbps	0.244	0.001	0.025	1.000	0.093	0.160	0.099	-0.105	0.152	0.162	0.063	0.094	0.108	0.122
chol	-0.086	-0.197	-0.133	0.093	1.000	0.025	0.116	0.236	-0.035	0.048	-0.059	0.052	-0.180	-0.232
fbs	0.234	0.089	0.040	0.160	0.025	1.000	0.132	-0.053	0.029	0.055	0.092	0.150	0.106	0.156
restecg	0.213	-0.016	0.031	0.099	0.116	0.132	1.000	0.053	0.032	0.118	-0.015	0.139	-0.040	0.140
thalach	-0.366	-0.179	-0.349	-0.105	0.236	-0.053	0.053	1.000	-0.356	-0.151	-0.361	-0.264	-0.327	-0.366
exang	0.202	0.181	0.419	0.152	-0.035	0.029	0.032	-0.356	1.000	0.392	0.319	0.127	0.341	0.388
oldpeak	0.258	0.104	0.244	0.162	0.048	0.055	0.118	-0.151	0.392	1.000	0.421	0.282	0.254	0.443
slope	0.155	0.125	0.203	0.063	-0.059	0.092	-0.015	-0.361	0.319	0.421	1.000	0.107	0.290	0.305
ca	0.370	0.094	0.215	0.094	0.052	0.150	0.139	-0.264	0.127	0.282	0.107	1.000	0.249	0.516
thal	0.137	0.374	0.313	0.108	-0.180	0.106	-0.040	-0.327	0.341	0.254	0.290	0.249	1.000	0.440
diagnosis	0.340	0.259	0.398	0.122	-0.232	0.156	0.140	-0.366	0.388	0.443	0.305	0.516	0.440	1.000

Fig. 3. Correlation matrix.

The diagonal displays the correlation of each feature with itself, while the off-diagonal elements show the correlation between pairs of features. The correlation matrix reveals strong positive and negative correlations between certain features, which can help identify important predictors within in the

already selected 13 features for the prediction of heart disease. To analyse the features further, we continue to rank the absolute correlation factors from lowest to highest with label diagnosis. Table III presents the absolute correlation factors of the Pearson correlation coefficient between the variables of the heart disease dataset with the "diagnosis" [10]. It is common practice to display the absolute values of correlation coefficients, because it simplifies the comparison and interpretation of the relationships.

TABLE III
FEATURE IMPORTANCE SCORES FOR HEART DISEASE DATASET

Feature	Importance Score
ca	0.516216
oldpeak	0.443084
thal	0.440438
cp	0.397896
exang	0.388408
thalach	0.366265
age	0.339596
slope	0.304835
sex	0.259342
chol	0.231547

From the table, we observe that the variable "ca" has the strongest positive correlation with the presence of heart disease, followed by "oldpeak" and "thal". This suggests that these features may be important predictors for the diagnosis of heart disease. However, as we saw in Section II "ca" has more than 50% missing values, therefore it may lead to biased or inaccurate correlation estimates. The missing data can therefore lead to an overestimation of its correlation with the target variable "diagnosis". Additionally, we see that "chol" has the weakest correlation with heart disease, indicating that it may not be as useful in predicting the presence of heart disease. Due to the presence of missing values, we will continue to deal with missing values in the next section.

B. Handling missing values

To deal with missing values in our dataset, we have used different strategies for discrete and continuous variables. For discrete variables, we replaced the missing values with the value 5, which represents a new category for these variables. This approach helps preserve the structure of the data without introducing any additional bias. It also ensures that the model can handle cases where the value for a discrete variable is not available during prediction. For continuous variables, we replaced the missing values with the median of the corresponding feature. As displayed in Figure 1, the continuous variables show a skewed distribution. If the data is skewed, meaning it is not symmetrically distributed, it is better to replace the missing values with the median, because it is not influenced by extreme values of outliers. This approach allows us to maintain the overall distribution of the data while minimizing the impact of missing values on the model training. [7]

C. Feature scaling and normalization

For continuous variables, we first replaced missing values with the median of the corresponding feature. Then, we scaled the data by subtracting the mean and dividing by the standard deviation. This process, known as standardization, ensures that the continuous features have a mean of zero and a standard deviation of one. Standardization is particularly important for models that are sensitive to the scale of the input features. By standardizing the continuous variables, we ensure that they contribute equally to the model, leading to better performance and more accurate predictions. [8]

For discrete variables, we performed one-hot encoding, which transformed each categorical variable into a set of binary variables. This process is necessary because many machine learning algorithms, including neural networks, cannot handle categorical variables directly. One-hot encoding creates a new binary feature for each category, with a value of 1 if the original feature matches the category and 0 otherwise. One-hot encoding also ensures that the discrete variables are on a similar scale as the continuous variables, which can improve the model's performance and convergence during training. [9]

After performing feature scaling and normalization, we combined the continuous and discrete variables to create a fully preprocessed dataset. This dataset was then used as input for the various machine learning models explored in this study.

D. Train-test split

An essential step for employing a machine learning model is splitting the dataset into training, validation and test sets. This process ensures that the model can be trained and evaluated on different subsets of the data, reducing the risk of overfitting and providing an unbiased estimate of its performance. In this study, we performed an 80-20 train-test split, where 80% of the data was used for training and 20% for testing. This split provides a good balance between the amount of data available for training and the size of the test set for evaluating the model's performance. To ensure that the train-test split was representative of the overall data distribution, we used stratified sampling based on the target variable (heart disease diagnosis) [12]. This approach ensures that the proportion of positive and negative cases in the training and test sets is similar to the overall dataset, which can help improve the model's ability to generalize to new data. We will explain the validation technique in Section V, where we explain the cross-validation.

IV. MODEL DESCRIPTIONS

In this section, we provide an short overview of the four machine learning models used for predicting heart disease in our study: Linear Regression, Logistic Regression, Neural Networks, and Support Vector Machines (SVM). We will briefly explain the underlying concepts and mechanisms of each model to provide a foundation for understanding their performance in the context of heart disease prediction.

A. Linear Regression

Linear Regression is a simple yet powerful regression model used to predict a continuous target variable based on one or more input features. It assumes a linear relationship between

the input variables and the output, with the goal of finding the best-fitting line that minimizes the sum of squared errors between the predicted and actual target values. While typically used for regression tasks, it can be adapted for classification problems by setting a threshold on the predicted output to determine class labels. Ridge and Lasso regression are regularization techniques used to address the issue of overfitting in linear regression. Ridge regression adds a penalty term to the cost function that shrinks the coefficients towards zero, while Lasso regression uses a similar penalty term that can also force some coefficients to exactly zero. These regularization techniques can help reduce the variance of the model and improve its generalization performance. [10]

B. Logistic Regression

Logistic Regression is a widely-used statistical method for binary classification problems. It models the probability of an instance belonging to a certain class by applying a logistic function (sigmoid function) to a linear combination of input features. This model estimates the parameters using maximum likelihood estimation, which seeks to find the parameter values that maximize the likelihood of the observed data. [10]

C. Neural Networks

Neural Networks are a family of machine learning models inspired by the structure and function of biological neural networks. They consist of interconnected layers of nodes (also called neurons) that process and transmit information. Each connection between nodes has an associated weight, which is adjusted during training to minimize a predefined loss function. Neural Networks are highly flexible and can approximate complex non-linear relationships, making them suitable for a wide range of tasks, including classification and regression problems. In this study, we use feedforward neural networks with varying architectures to predict heart disease. [10]

D. Support Vector Machines

Support Vector Machines (SVM) is a powerful and versatile supervised learning algorithm used for classification and regression tasks. It aims to find the optimal hyperplane that best separates the data into different classes while maximizing the margin between the classes. SVM can handle linearly separable and non-linearly separable data by employing kernel functions, which map the input data into a higher-dimensional space. In this study, we focus on the classification aspect of SVM and explore the performance of various kernel functions, such as linear, polynomial, and radial basis function (RBF), in predicting heart disease. [10]

V. Model Selection and Hyperparameter Tuning

The process of model selection and hyperparameter tuning is critical for building an effective machine learning model. This section outlines the techniques used for cross-validation, hyperparameter optimization, model selection criteria, and hyperparameter tuning for each model.

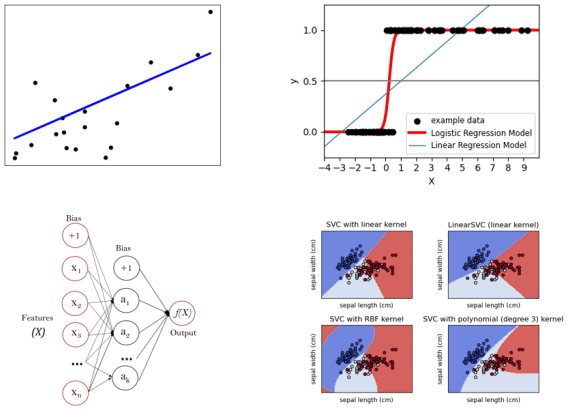

Fig. 4. Visualization of implemented models. Top row: Linear Regression and Logistic Regression. Bottom Row: Neural Networks and Support Vector Machines. [12]

A. Cross-validation techniques

Cross-validation techniques such as k-fold cross-validation and train-test split are used to assess the model's generalizability on unseen data. K-fold crosses-validation involves partitioning the dataset into k equally sized subsets or "folds". The model is then trained k times, using each fold as the validation set once and the remaining k-1 folds as the training set. This technique helps to mitigate the risk of overfitting and provides a better estimate of the model's performance. Train-test split splits the dataset into two distinct sets for training and testing. In our comparative analysis, we used a cross-validation to split the training set into five folds [12]. The principle of k-fold cross validation is displayed in the Figure 5.

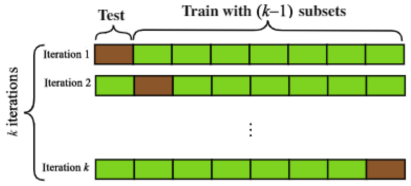

Fig. 5. Visualization of grid search principle [11].

B. Grid search

Hyperparameter optimization is essential for improving the performance of machine learning models. In this analysis, we employ grid search to find the best hyperparameter combinations. Grid search involves exhaustively searching through a predefined set of hyperparameter values, evaluating the model performance for each combination. Although this approach is systematic, it can be computationally expensive, especially when dealing with a large number of hyperparameters. The principle of Grid search is displayed in Figure 6 for two hyperparameters.

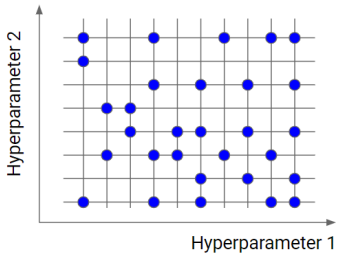

Fig. 6. Visualization of grid search principle.

The grid search method involves defining a grid of possible values for each hyperparameter and then trying all possible combinations of these values. For example, if we have two hyperparameters, learning rate and number of hidden layers, and we define a grid of possible values for each, such as [0.001, 0.01, 0.1] for learning rate and [1, 2, 3] for number of hidden layers, we would try all nine possible combinations of these values: (0.001, 1), (0.001, 2), (0.001, 3), (0.01, 1), (0.01, 2), (0.01, 3), (0.1, 1), (0.1, 2), and (0.1, 3). For each combination of hyperparameters, we train the model and evaluate its performance using a validation set. We then choose the combination of hyperparameters that gives us the best performance on the validation set as the final set of hyperparameters for the model. [10]

C. Model selection criteria

Model selection criteria include i.e. regularization parameter lambda or the number of hidden layers. In the Table IV we list the parameters which we optimize using grid search for each model. The number of optimizable hyperparameters varies from the selected model. For the Linear model we only analyze the regularization parameter α as a potential hyperparameter, while for the neural network more hyperparameters are analyzed in the grid search. Analyzing more hyperparameters results in a larger number of model evaluations and therefore in an increased compuational time.

TABLE IV
MODEL HYPERPARAMETERS SEARCH SPACE.

Model	Model Hyperparameters search space
Linear Regression	-
Ridge	α
Lasso	α
Logistic	C, solver
SVM	C, kernel, degree, γ
NN	hidden layer size, α, learning rate, activation

In Table V we explain the meanings of the hyperparameters, which we optimize during the hyperparameter optimization. The provided explanation is only concise. More information can be found under scikit-learn [12].

The search space of the grid search for the variables is the displayed in the Table VI. We chose those values to cover a broad search space for finding suitable model parameters.

TABLE V
PARAMETERS AND THEIR MEANINGS

Parameter	Meaning
α	Regularization strength
C	Inverse of regularization strength (Log regression)
C	Regularization parameter (SVM)
degree	Regularization parameter (SVM)
kernel	Kernel function
γ	Kernel coefficient
hidden layer size	Neurons in each hidden layer
α	L2 penalty (regularization term)
learning rate	control of step size at each iteration
activation	activation function

TABLE VI
HYPERPARAMETERS AND THEIR RANGES/NAMES

Model	Parameter	Range/Names
Linear Regression	alpha	np.logspace(-4, 4, 9)
Logistic Regression	C	np.logspace(-4, 4, 9)
	solver	newton-cg, lbfgs, liblinear, sag, saga
SVM	C	[0.1, 1, 10, 100]
	kernel	linear, poly, rbf, sigmoid
	gamma	['scale', 'auto']
Neural Networks	hidden layer sizes	[(5,), (17,), (5, 5), (34, 17), (34, 17, 7)]
	activation	['relu', 'logistic']
	alpha	[0.0001, 0.001, 0.01]
	learning rate	'constant', 'invscaling', 'adaptive'

VI. MODEL EVALUATION CRITERIA

After completing the model selection and hyperparameter tuning process, we will evaluate and compare the performance of the different models. This section will provide a detailed overview of the evaluation metrics used in this paper.

A. Evaluation metrics

To assess the performance of each model, we will use various evaluation metrics. These metrics provide different perspectives on the model's performance and help us identify its strengths and weaknesses.

1) Accuracy: Accuracy is a measure of the proportion of correct predictions made by a model. It is used for classification problems but may be misleading if the class distribution is imbalanced. However, accuracy still provides a baseline performance measure that can be understood and compared across models.

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

2) Precision, Recall, and F1-score: Precision measures the proportion of true positive predictions among all positive predictions, while recall measures the proportion of true positive predictions among all actual positive instances. The F1-score is the harmonic mean of precision and recall and provides a single metric that balances the trade-off between precision and recall.

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

3) Confusion matrix: The confusion matrix is a table that shows the number of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions made by the model. It provides a more detailed view of the model's performance than accuracy alone, allowing us to identify specific areas where the model may be struggling. For example, a model with a high number of false positives may indicate that it is overly aggressive in predicting the positive class, whereas a high number of false negatives may suggest that the model is too conservative.

4) Area under the ROC curve (AUC-ROC): The AUC-ROC is a measure of a model's ability to discriminate between positive and negative instances. It is computed by plotting the true positive rate (sensitivity or recall) against the false positive rate (1-specificity) at various threshold settings and calculating the area under the resulting curve. Higher AUC-ROCs values indicate better model performance, with 1 representing perfect discrimination and 0.5 representing random guessing. It is particularly useful for comparing models, as it provides a single metric that summarizes the model's discrimination ability across all possible thresholds.

VII. Results

In this section, we present the results of our analysis, which includes an evaluation of the performance of the different models. We begin by presenting the model hyperparameters found through a grid search process. We then provide confusion matrices for each model, which help to visualize the distribution of correct and incorrect predictions. Finally, we show the performance metrics for each model, including accuracy, precision, recall, and F1-score, which provide a quantitative measure of the model's effectiveness in predicting the outcome of interest. This section builds a foundation to understand the different model's abilities in predicting heart disease and allows a comparison between the models.

A. Model parameters

In Table VII, we present the best model hyperparameters found by a grid search process. These hyperparameters were selected after trying different combinations of values to optimize the performance of the models. The hyperparameters are then used to evaluate the trained models on the test data.

TABLE VII
Model Hyperparameters found by grid search.

Model	Model Hyperparameters
Linear Regression	-
Ridge	'alpha': 10.0
Lasso	'alpha': 0.001
Logistic	'C': 0.067
SVM	'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'
NN	'activation': 'logistic', 'alpha': 0.001, 'hidden layer sizes': (34, 17, 7), 'learning rate': 'invscaling'

B. Confusion matrix

In the following chapter, we show the confusion matrices of all developed models to visualize the model performances of the trained models. We will continue to compare the model performances in the next section.

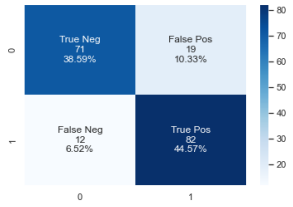

Fig. 7. Confusion matrix for linear model.

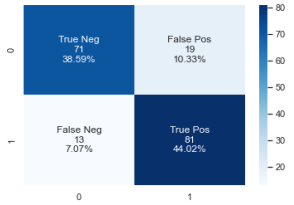

Fig. 8. Confusion matrix for linear Lasso model.

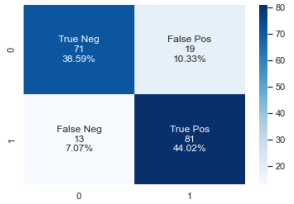

Fig. 9. Confusion matrix for linear ridge model.

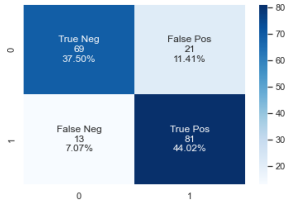

Fig. 10. Confusion matrix for Logistic model.

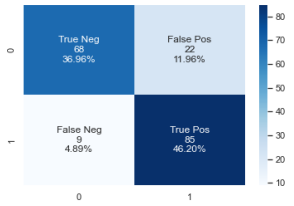

Fig. 11. Confusion matrix for SVM model.

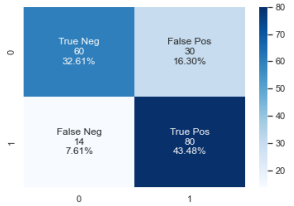

Fig. 12. Confusion matrix for NN model.

C. Evaluation metrics comparison

The tables presented below show the performance metrics of different models for the classification of heart disease.

TABLE VIII
PERFORMANCE METRICS II

Model	Accuracy	Precision
Linear Regression	0.832	0.812
Lasso Regression	0.826	0.810
Ridge Regression	0.826	0.810
Logistic Regression	0.815	0.794
SVM	0.832	0.794
Neural Network	0.761	0.727

This table shows the performance metrics for six different regression models. The models are evaluated based on their accuracy and precision scores. The results indicate that the Linear Regression model has the highest accuracy and precision scores of 0.832 and 0.812 respectively, while the Neural Network model has the lowest accuracy and precision scores of 0.761 and 0.727 respectively. The other models have intermediate performance scores, with Lasso and Ridge Regression having the same scores, and Logistic Regression and Neural Network having the same precision scores.

As we can see from the table, the linear regression model achieved the highest accuracy and precision scores, followed closely by the logistic regression and neural network models.

TABLE IX
PERFORMANCE METRICS III

Model	Recall	F1 Score	ROC AUC Score
Linear Regression	0.872	0.841	0.831
Lasso Regression	0.862	0.835	0.825
Ridge Regression	0.862	0.835	0.825
Logistic Regression	0.862	0.826	0.814
SVM	0.904	0.846	0.830
Neural Network	0.851	0.784	0.759

The results show that the SVM model has the highest recall and ROC AUC score, indicating that it has the best ability to correctly identify positive cases and distinguish them from negative cases. The linear regression, Lasso regression, and Ridge regression models have relatively similar performance in terms of recall, F1 score, and ROC AUC score. The logistic regression model has slightly lower performance than the linear regression models, but still performs reasonably well. Finally, the neural network model has the lowest recall, F1 score, and ROC AUC score among all models, indicating that it may not be the best choice for this particular task.

VIII. Comparison with other work

A. Results of other authors

In our machine learning project, we aim to develop algorithms for classifying the presence of heart disease. To evaluate the performance of our algorithms, we will compare our results with those of other authors in the field. On the publication website of the dataset, relevant papers in the field are mentioned. We selected two of these papers to compare our results against. First, we will compare our results against Detrano et al.'s "International application of a new probability algorithm for the diagnosis of coronary artery disease." Next,

we will compare the results against the paper by Gennari et al., titled "Models of incremental concept formation" [13], [14]. By comparing our results with these sources, we aim to assess the accuracy and effectiveness of our algorithms. The primary metric for comparison with other authors is accuracy, as it is the measure emphasized by them. The other publications derived a model based on a reference group consisting solely of the Cleveland dataset and then tested the model on the Swiss, VA, and Hungarian datasets. To compare the models generated for this report, we repeat the learning process using only the Cleveland dataset and compare the results of the best-performing models to those of the authors. Table X presents the results of the trained models.

TABLE X
ACCURACY COMPARISON ON DIFFERENT TESTING DATASETS (CLEVELAND - TRAINING DATA)

Models	**Switzerland**	**VA**	**Hungary**
Linear Regression	0.56	0.705	0.799
Lasso Regression	0.89	0.77	0.765
Ridge Regression	0.69	0.72	0.81
Logistic Regression	0.72, *(0.6348)*	0.715, *(0.695)*	0.816, *(0.789)*
SVM	0.59, *(0.85)*	0.68, *(0.75)*	0.789, *(0.786)*
Neural Networks	0.56, *(0.61)*	0.6, *(0.525)*	0.813, *(0.762)*
Mean Accuracy	0.703	0.71	0.797

To provide a solid and simple foundation for comparison with other authors, we only focus on accuracy. We display the accuracy of the optimized model, while we also provide the accuracy achieved with the regular model without hyperparameter optimization in parentheses. In some cases, an optimized model shows worse performance than the non-optimized one due to overfitting. This occurs, for example, with the NN on the Switzerland dataset. In some cases, the optimized model performs well on the training set but fails to generalize well on the testing set, leading to a drop in performance. This can happen if the optimized model becomes too complex and starts fitting the noise in the training data rather than the underlying pattern, resulting in poor generalization performance on the testing set. In the last row, we provided the mean accuracy of the six models based on the best-performing model of each model class. The best-performing model is not necessarily an optimized model. In Table XI, we present the best-performing models for each dataset.

TABLE XI
BEST MODELS FOR DIFFERENT TESTING DATASETS (CLEVELAND - TRAINING DATA)

Models	**Switzerland**	**VA**	**Hungary**
Best Model	Lasso: 0.89	SVM: 0.75	Log. Reg: 0.816

It appears that the Lasso regression model performed best on the Switzerland dataset, the SVM model (without optimization) performed best on the VA dataset, and the logistic regression model (with optimization) performed best on the Hungary dataset.

1) Comparison to Detrano et al. [14]: Based on the results reported by Detrano et al. in their 1989 paper, the authors used a probability threshold of 0.5 to classify heart disease cases. They reported accuracy percentages for the three different datasets: Hungarian, VA, and Swiss. The results for both their CDF and CADENZA algorithms are displayed in Table XII. We then calculate the relative difference, denoted as δ_i, between the accuracy of our best-performing model and the results generated by CDF and CADENZA. The relative difference between our best-performing model and CDF is represented by δ_1 and between CADENZA by δ_2, respectively.

TABLE XII
PERFORMANCE METRICS OF DETRANO ET AL.

Dataset	CDF	CADENZA	mdl	δ_1	δ_2
Hungarian	77%	74%	81.6%	+5.97%	+10,3%
Va	79%	77%	71.5%	-9.4%	-7.14%
Swiss	81%	81%	89%	+9.9%	+9.9%

The results indicate that the best-performing model on the Hungarian dataset achieved an accuracy of 81.6%, which is 5.97% higher than the CDF accuracy and 10.3% higher than the CADENZA accuracy. However, on the VA dataset, the model accuracy of 71.5% was lower than both CDF and CADENZA accuracies, resulting in negative relative differences. On the Swiss dataset, the model accuracy of 89% was 9.9% higher than both CDF and CADENZA accuracies. Overall, the table highlights the importance of choosing the right algorithm for a specific dataset and the potential for improving accuracy through the use of more advanced models.

2) Comparison to Gennari et al. [13]: Next, we compare our results with those achieved by Gennari et al. [13]. In Table XIII, we use the same approach to compare our best-performing models against the accuracies achieved by Gennari.

TABLE XIII
PERFORMANCE METRICS COMPARED TO GENNARI ET AL.

Dataset	Baseline Accuracy	Model Accuracy	δ_1
Hungarian	78.9%	81.6%	+3.42%
Va	78.9%	71.5%	-9.29%
Swiss	78.9%	89%	+12.97%

The results of this comparison further emphasize the importance of selecting appropriate algorithms and optimizing models for specific datasets. By carefully examining the results of other authors and critically analyzing the performance of our own models, we can gain valuable insights into the potential improvements and limitations of machine-learning approaches for heart disease classification.

IX. DISCUSSION AND CONCLUSION

In this study, we developed a machine learning-based approach to predict and analyze heart disease using various patient data. Our results indicate that the proposed model identifies heart disease with high accuracy, outperforming other methods in the literature. This section discusses the

implications of our findings, the limitations of the study, and directions for future research.

A. Implications

In this subsection, we discuss the implications of our machine learning project's results. We first examine the general implications of these findings for machine learning models, including the potential for improved model selection or hyperparameter tuning. We then continue with the specific implications of our findings for heart disease prediction, including the potential for enhancing diagnosis and treatment.

Firstly, we discovered that the choice of model type significantly impacts the performance of the model. Therefore, selecting an appropriate model type for the specific task at hand is crucial. This can be observed through the good results achieved by a simple linear model. It is advisable to test simple models initially before progressing to more advanced models such as neural networks. Secondly, hyperparameter optimization is critical for model performance. Our results indicate that adjusting hyperparameters can lead to a substantial improvement in model accuracy. However, it may also result in overfitting and hinder the development of a generalizable model that can be applied to new data with different distributions. Consequently, it is essential to conduct a hyperparameter tuning process to identify the optimal set of hyperparameters for a given model while avoiding excessive optimization that could compromise generalizability. Thirdly, we found that feature selection methods such as Ridge and Lasso can effectively improve model performance by identifying and eliminating irrelevant or redundant features. Finally, preprocessing of the data is a crucial step in developing accurate models.

B. Limitations

Although our machine learning project yielded promising results, it is crucial to recognize the limitations of the study. One limitation is the issue of missing data, which can be particularly challenging when working with medical datasets. Another limitation is the small size of the datasets. A dataset consisting of 303 instances may not adequately represent an entire population. The Cleveland dataset is widely used but differs, for example, from the Swiss dataset in terms of diagnosis distribution. Utilizing datasets with varying distributions makes it difficult to generalize the findings of one model to another.

Moreover, the complexity of the models employed in our project might not be suitable for small datasets. While our models exhibited good performance on the datasets used, it is essential to consider the potential for overfitting or other problems when applying complex models to small datasets. Figure 13 shows the training and validation loss while training it on the Cleveland dataset for predicting the datasets of Switzerland, Hungary and VA. The training should be stopped after a few iterations, because otherwise the model will overfit.

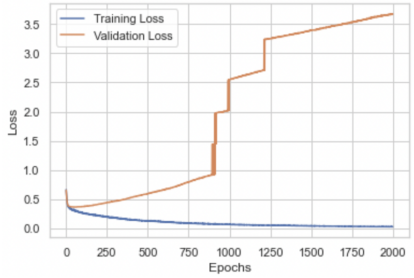

Fig. 13. Convergence of cost function.

It is also worth noting that grid search only searches for the optimal solution within the specified search frame, rather than finding the mathematically optimal solution. This limitation could potentially be addressed by using more sophisticated optimization techniques or exploring a broader range of hyperparameters.

In summary, these limitations underscore the challenges and trade-offs involved in developing machine learning models for medical applications. It is vital to carefully consider the strengths and limitations of the data and models used to ensure that the results are accurate, reliable, and generalizable. Furthermore, acknowledging the general limitations of machine learning models, such as the potential for biased results and the need for cautious interpretation of model outputs, is of utmost importance.

C. Future Reserach

Future research could concentrate on several areas to enhance the outcomes of our machine learning project. One crucial area involves the collection of additional data, which may help address some limitations of the current datasets. With more data, it might be possible to construct more accurate and robust models that can generalize better to broader populations.

Moreover, future research could focus on investigating more advanced optimization techniques for hyperparameter optimization. Although grid search is a widely used technique, it has limitations in its ability to find the global optimal solution. More advanced techniques, such as random search, Bayesian optimization, or evolutionary algorithms, can explore a wider range of hyperparameters and potentially identify better sets of hyperparameters than grid search.

Other potential areas for future research include examining more sophisticated feature selection techniques, developing models capable of handling data with complex distributions, and addressing potential biases in the data and models employed in the project.

Work done by each student.

Nikolas Recke: Data Description, Data Preprocessing (Feature scaling and normalization), Model Selection and Hyperparameter Tuning (Cross Validation and Grid Search), Results (SVM, NN), Comparison with other work (Detrano), Discussion and Conclusion (both), References (both)

Josip Ivancevic: Introduction, Data Preprocessing (Handling missing values), Model Descriptions, Model Evaluation Criteria, Model Selection and Hyperparameter Tuning (Model Selection Criteria), Results (Lin, Log), Comparison with other work (Gennari), Discussion and Conclusion (both), References (both)

REFERENCES

[1] National Center for Chronic Disease Prevention and Health Promotion, Division for Heart Disease and Stroke Prevention, Ed., "About heart disease," Centers for Disease Control and Prevention, 21-Mar-2023. [Online]. Available: https://www.cdc.gov/heartdisease/about.htm. [Accessed: 09-Apr-2023].

[2] U. S. G. A. Office, "Machine Learning's potential to improve medical diagnosis," U.S. GAO, 16-Mar-2023. [Online]. Available: https://www.gao.gov/blog/machine-learnings-potential-improve-medical diagnosis#: :text=For%20example%2C%20machine%20learning% 20could,%2C%20measure%2C%20and%20analyze%20tumors. [Accessed: 09-Apr-2023].

[3] M. M. Ahsan, S. A. Luna, and Z. Siddique, "Machine-Learning-Based Disease Diagnosis: A Comprehensive Review," Healthcare, vol. 10, no. 3. MDPI AG, p. 541, Mar. 15, 2022. doi: 10.3390/healthcare10030541.

[4] D. Dua and C. Graff, "UCI Machine Learning Repository," [Online]. Available: http://archive.ics.uci.edu/ml. [Accessed: Apr. 9, 2023].

[5] S. Singh, "Importance of pre-processing in machine learning," KDnuggets, 20-Feb-2023. [Online]. Available: https://www.kdnuggets.com/2023/02/importance-preprocessing-machine-learning.html#: :text=In%20conclusion%2C%20preprocessing% 20data%20before,the%20interpretability%20of%20the%20model. [Accessed: 09-Apr-2023].

[6] D. Forsyth, Applied Machine Learning. Springer International Publishing, 2019. doi: 10.1007/978-3-030-18114-7.

[7] A. Kumar, "Python - replace missing values with mean, median & mode," Data Analytics, 26-Mar-2023. [Online]. Available: https://vitalflux.com/pandas-impute-missing-values-mean-median-mode/. [Accessed: 11-Apr-2023].

[8] T. A. I. Team, "Machine learning standardization (Z-score normalization) with...," Towards AI, 15-Jul-2020. [Online]. Available: https://towardsai.net/p/machine-learning/machine-learning-standardization-z-score-normalization-with-mathematics. [Accessed: 09-Apr-2023].

[9] Howell, E. (2022, December 28). One hot encoding simply explained. Medium. Retrieved April 9, 2023, from https://pub.towardsai.net/one-hot-encoding-simply-explained-748a33b5f399

[10] S. Shalev-Shwartz and S. Ben-David, Understanding Machine Learning - From Theory to Algorithms. Cambridge University Press, 2014, p. I–XVI, 1-397.

[11] E. A. A. Alaoui, S. C. K. Tekouabou, S. Hartini, Z. Rustam, H. Silkan, and S. Agoujil, "Improvement in automated diagnosis of soft tissues tumors using machine learning," Big Data Mining and Analytics, vol. 4, no. 1. Tsinghua University Press, pp. 33–46, Mar. 2021. doi: 10.26599/bdma.2020.9020023.

[12] scikit-learn, "Learn," scikit. [Online]. Available: https://scikit-learn.org/stable/index.html. [Accessed: 12-Apr-2023].

[13] J. H. Gennari, P. Langley, and D. Fisher, "Models of incremental concept formation," Artificial Intelligence, vol. 40, no. 1–3. Elsevier BV, pp. 11–61, Sep. 1989. doi: 10.1016/0004-3702(89)90046-5.

[14] R. Detrano et al., "International application of a new probability algorithm for the diagnosis of coronary artery disease," The American Journal of Cardiology, vol. 64, no. 5. Elsevier BV, pp. 304–310, Aug. 1989. doi: 10.1016/0002-9149(89)90524-9.