

**Final project – Machine learning models for predicting heart disease in patients**

Machine learning and data mining course: 364-2-1651

Lecturer: Prof. Boaz Lerner

**Submitted by:**

Name: Inbal Karibian ID: 315861773

Name: Adi Yogev Golbari ID: 307849869

Name: Arad Peleg ID: 206068801

תמונה שמכילה רהיטים, טקסט, שולחן, לבוש

התיאור נוצר באופן אוטומטי

**Abstract**

Heart disease, one of the leading causes of death worldwide, accounts for approximately 17.9 million deaths annually and incurs significant healthcare costs, such as $239.9 billion per year in the United States alone. Accurate diagnosis is crucial due to the complexity of the disease, influenced by genetic, lifestyle, and physiological factors. In this project, we used data from over 1,190 patient records and applied various machine learning models to predict heart disease. The Random Forest model achieved the highest accuracy at 97.48%, making it the most reliable. Despite hyperparameter tuning, the Decision Tree, Random Forest, and XGBoost models did not improve in accuracy. However, the Gradient Boosting model showed some improvement but did not surpass Random Forest. Our approach aims to enhance the predictive accuracy of heart disease diagnosis, providing valuable insights that can support healthcare professionals in identifying high-risk patients and improving treatment strategies.

**Table of Contents**

[1. Business Understanding 5](#_Toc170049128)

[1.1. Healthcare Industry Overview 5](#_Toc170049129)

[1.2. Strategic Goal 5](#_Toc170049130)

[1.3. Current Situation 5](#_Toc170049131)

[1.4. Solution 5](#_Toc170049132)

[2. Data understanding 6](#_Toc170049133)

[2.1. Data Collection 6](#_Toc170049134)

[2.2. Data Explanation 6](#_Toc170049135)

[2.3. Data Exploration 6](#_Toc170049136)

[3. Data Preparation 9](#_Toc170049137)

[3.1. Data Selection 9](#_Toc170049138)

[3.2. Data Cleaning 9](#_Toc170049139)

[3.3. Filling Missing Values 9](#_Toc170049140)

[3.4. Data Construction 9](#_Toc170049141)

[3.5. Data Formatting 10](#_Toc170049142)

[4. Modeling 11](#_Toc170049143)

[4.1. Models Used 11](#_Toc170049144)

[4.2. Importance Of Features 11](#_Toc170049145)

[4.3. Hyper Parameter Tuning 12](#_Toc170049146)

[5. Evaluation 13](#_Toc170049147)

[5.1. Model Evaluation – Base Models 13](#_Toc170049148)

[5.2. Model Evaluation – Hyper Tuned Models 14](#_Toc170049149)

[6. Discussion and Conclusions 14](#_Toc170049150)

[7. Bibliography 15](#_Toc170049151)

[8. Appendices 16](#_Toc170049152)

**Table of Figures**

[Figure 1 - The distribution of each nonbinary feature. 7](#_Toc170062575)

[Figure 2 - The binary feature distribution. 7](#_Toc170062576)

[Figure 3 - Showing scatterplot graphs.. 7](#_Toc170062577)

[Figure 4 - Correlation matrix. 8](#_Toc170062578)

[Figure 5 - cholesterol feature after mean and log normalization. 9](#_Toc170062579)

[Figure 6 - new correlation graph, with new features. 10](#_Toc170062580)

[Figure 7 - Feature importance of each model. 11](#_Toc170062581)

[Figure 8 - evaluating models using different scoring metrics. 13](#_Toc170062582)

[Figure 9 - Descriptive Statistics 16](#_Toc170062583)

[Figure 10 - Attribute Description and Missing Values 16](#_Toc170062584)

**Table of Tables**

[Table 1 - features categories. 6](#_Toc170210341)

[Table 2 - Explanation of new feature construction. 10](#_Toc170210342)

[Table 3 - Hyper parameters used for each model. 12](#_Toc170210343)

[Table 4 - scores after optimization. 14](#_Toc170210344)

[Table 5 - Data variable explanation part 1. 16](#_Toc170210345)

[Table 6 - Data variable explanation part 2. 17](#_Toc170210346)

[Table 7 - Models we choose to use in this project and why. 17](#_Toc170210347)

**Table Of Abbreviations**

|  |  |
| --- | --- |
| World Health Organization | WHO |
| Coronary artery disease | CAD |
| Centers for Disease Control and Prevention | CDC |
| exploratory data analysis | EDA |
| Artificial intelligence | AI |
| Framingham Risk Score | FRS |
| Total Cardiovascular Risk Score | TCRS |
| Support Vector Machine | SVM |
| K-Nearest Neighbours | KNN |
| Multi-layer Perceptron | MLP |

# **1. Business Understanding**

### **1.1. Healthcare Industry Overview**

Heart disease, also known as cardiovascular disease, includes various conditions affecting the heart and remains the leading cause of death worldwide. According to the World Health Organization (WHO), cardiovascular diseases are responsible for approximately 17.9 million deaths each year, with coronary artery disease and cerebrovascular disease being the most common with 80% of the cases (Shah et al., 2020). The heart is vital for pumping blood, supplying oxygen, and nutrients to the body. If the heart malfunctions, other organs may also fail to function properly (Katarya & Meena, 2021).

For example, heart disease is the leading cause of death in the United States, with an annual cost of around $239.9 billion covering healthcare, medications, and lost productivity. Coronary artery disease (CAD), the most common type, killed 375,476 people in 2021, as reported by the United States government (Centers for Disease Control and Prevention – CDC, Link: [CDC](https://www.cdc.gov/heart-disease/data-research/facts-stats/index.html) ).

### **1.2. Strategic Goal**

In the healthcare industry, predicting heart disease is crucial for enhancing patient outcomes and reducing severe cardiac events. Accurate early diagnosis and intervention can significantly improve patient care. The primary goal is to use effective strategies and medical data to identify individuals at risk of heart disease, improving patient management, providing medical interventions, and ultimately reducing healthcare costs for individuals and the healthcare industry.

### **1.3. Current Situation**

Heart disease is influenced by genetic predisposition, personal habits (smoking, excessive alcohol and caffeine use, stress, physical inactivity), and physiological conditions (obesity, hypertension, high cholesterol, and pre-existing heart conditions) (Shah et al., 2020). Efficient, accurate, and early diagnosis is crucial for preventing fatalities. Demographic attributes such as age and sex can also affect individuals (Katarya & Meena, 2021).

According to (Soni et al., n.d.), the medical diagnosis of heart disease is complex and must be executed accurately and efficiently. Automating this process would be highly beneficial, as not all doctors have expertise in every subspecialty, and there is a shortage of specialists in some areas.

Computer-aided and machine learning techniques can detect hidden patterns in input datasets to build models, improving the speed and accuracy of patient predictions while significantly reducing costs (Jindal et al., 2021; Shah et al., 2020).

### **1.4. Solution**

The goal is to better understand the factors affecting heart diseases and to develop a data-based prediction machine learning tool for early diagnosis and individual risk assessment. This involves investigating the heart disease database through exploratory data analysis (EDA) and experimenting with classification algorithms for prediction. To achieve this, various machine learning models will be compared to determine which ones offer the highest accuracy and reliability for heart disease prediction.

# **2. Data understanding**

### **2.1. Data Collection**

The dataset is taken from Kaggle, it was created by combining five popular heart disease datasets with data obtained from the following locations: Cleveland, Hungary, Switzerland, and Long Beach, California. Additionally, it includes the Statlog dataset, which is related to AI heart disease projects. This comprehensive dataset integrates diverse patient information and medical records, providing a robust foundation for developing predictive AI models (Link: [Heart Disease Dataset](https://www.kaggle.com/datasets/mexwell/heart-disease-dataset)).

### **2.2. Data Explanation**

The data is merged with the most common 11 features from the different data sets that was mentioned before. resulting in a dataset with 1,190 observations. It was collected as a csv file; each row represents a single patient instance with an addition of target instance for each patient that when is 1 it means disease and when 0 it means no disease.

In order to build features that are based on real indicators from the world of medicine, our deeper understanding of the data and our convenience, we decided to divide the features into 6 categories:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Demographic | Symptom-related | Clinical Measurements | Electrocardiographic (ECG) Results | Exercise-related | Diagnostic Measures |
| * Age * Sex | * Chest Pain Type | * Resting Blood Pressure * Cholesterol * Fasting Blood Sugar | * Resting ECG | * Max Heart Rate * Exercise Angina | * Oldpeak * ST Slope |

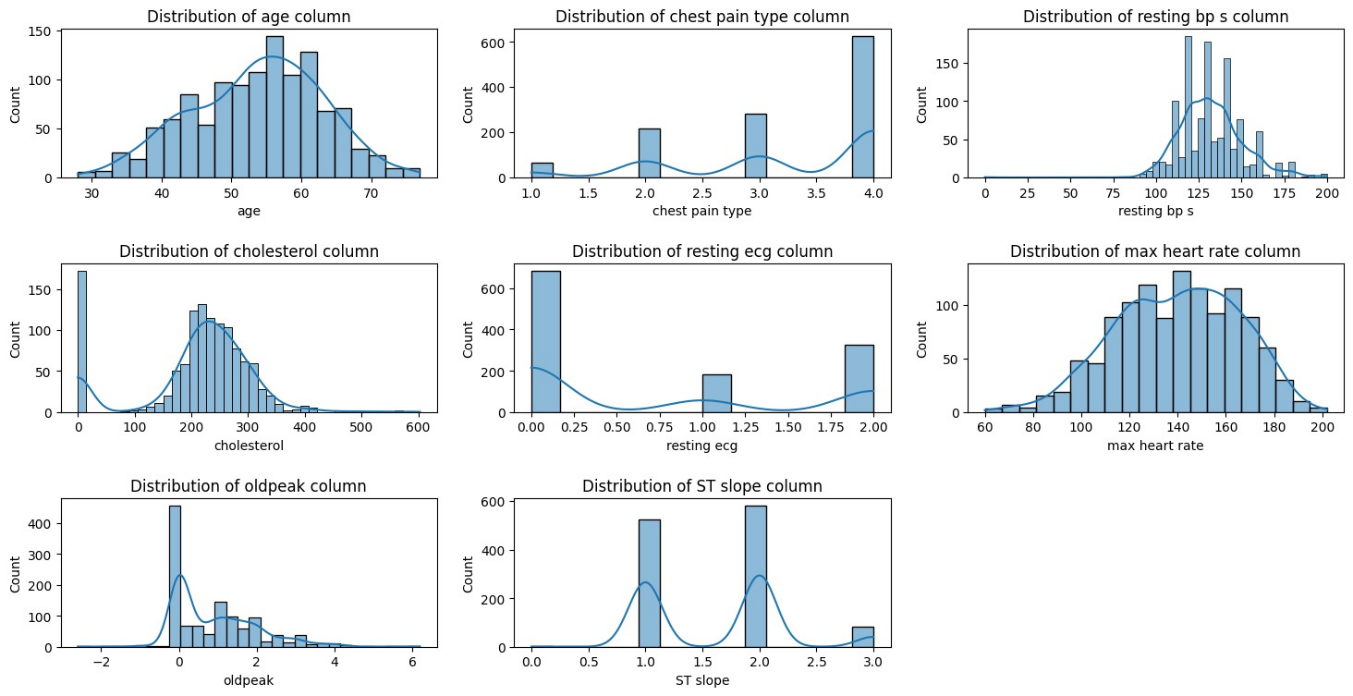
**Table 1 - features categories.**

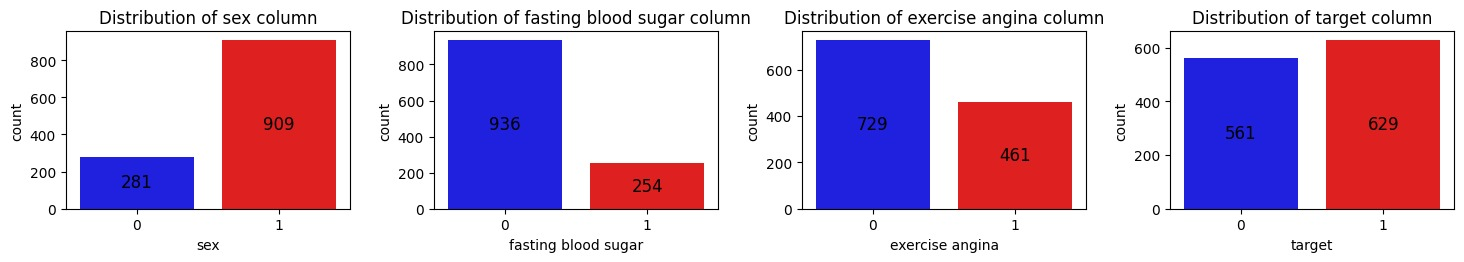
After we did that, we showed descriptive statistics for all features and attribute description (info, all data is numeric) for understanding values and variables within the data itself. both can be found in the appendices (Figure 9 and 10) at the end of the report. Additionally in the appendices you will find each feature and value explanation in the data.

### **2.3. Data Exploration**

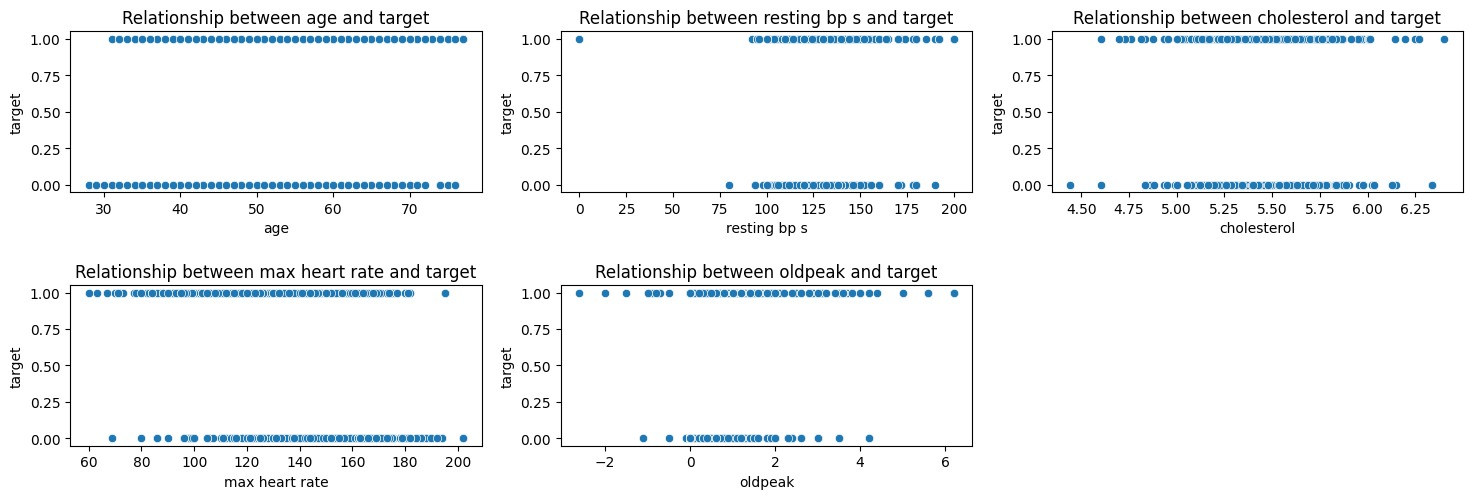
In terms of data balance, we can say our data is balanced because we know data is not balanced when A ratio between classes is 1.3 or higher (considered unbalanced). However, in this dataset, the ratio of 629 samples in class 1 (disease) to 561 samples in class 0 (no disease) is about 1.12 (629/561). This ratio is very close to the ideal 1:1 ratio balance, indicating that the dataset is well-balanced and suitable for training classification models without needing to use balancing methods for class imbalance.

As part of our data understanding process, we visualized the feature distributions and explored interesting relationships and correlations within the dataset. This analysis helped us understand how to handle the data more effectively and consider potential feature creation options. Below find some interesting conclusions (figures: 1,2,3 and 4):

**Figure 1 - The distribution of each nonbinary feature.**

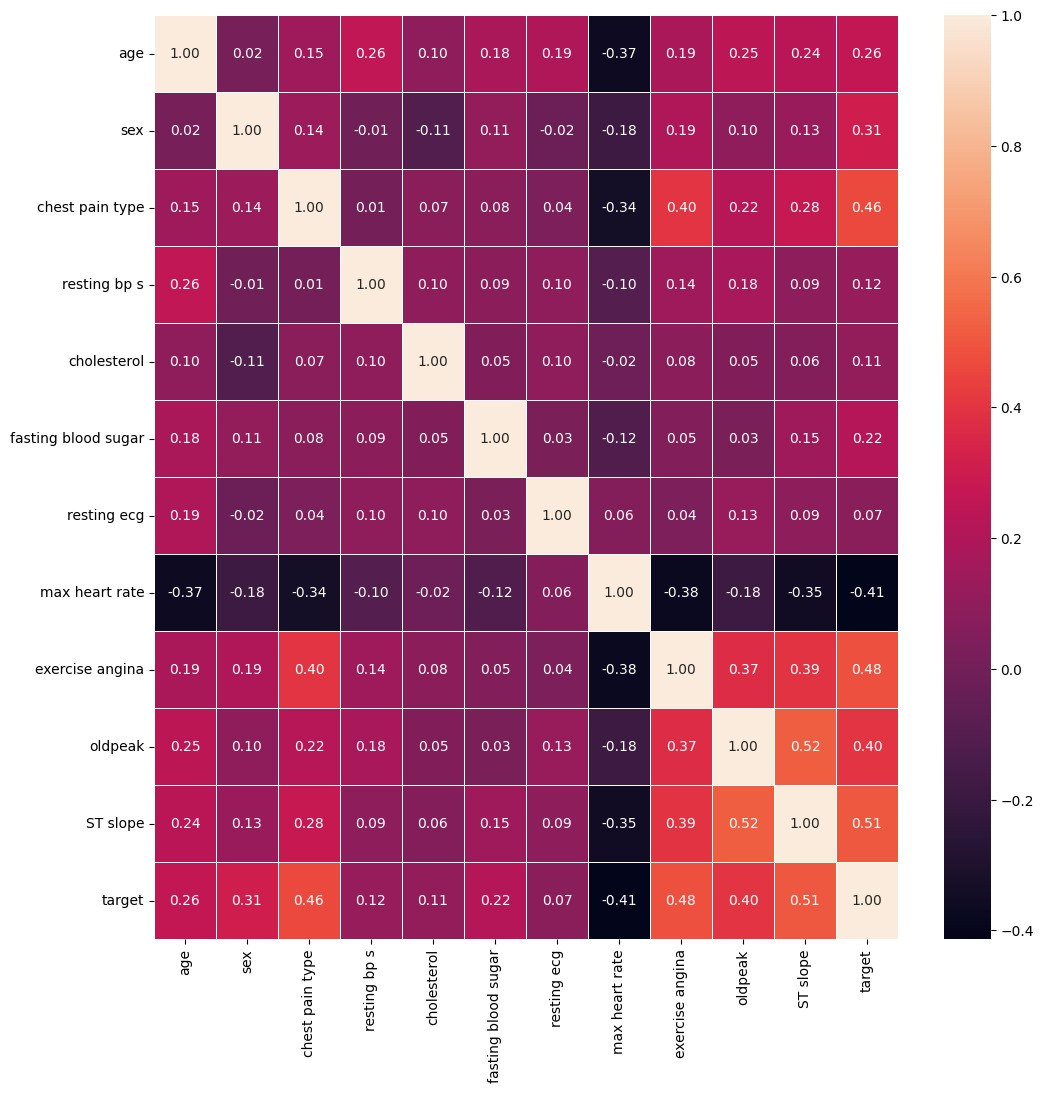
* The dataset likely focuses on middle-aged to older individuals with potential cardiovascular issues, 55-60.
* A significant portion of the individuals in the dataset have high blood pressure readings, 130-140 mmHg.
* Many individuals in the dataset exhibit elevated cholesterol levels that rages between 150 to 300 mg/dL, it important to note that its major risk factor for cardiovascular diseases and strokes (Link: [What Your Cholesterol Levels Mean](https://www.heart.org/en/health-topics/cholesterol/about-cholesterol/what-your-cholesterol-levels-mean)).
* Also, in cholesterol we can see that some patience didn’t have their cholesterol level recorded and have the value of zero we will show how we handled it in the data preparation stage.

**Figure 2 - The binary feature distribution.**

* In sex feature there are more male instances (male -1).
* in fasting blood sugar feature, more female suffers from it (female -0).

**Figure 3 - Showing scatterplot graphs. Columns with more than four values against target column to see if there is a trend for importance of features.**

* **Max Heart Rate and Target** - Higher maximum heart rates are strongly associated with the target condition, suggesting that individuals with higher max heart rates are more likely to test positive for the condition.
* **Oldpeak and Target** - Lower oldpeak values are mostly linked to the target condition, indicating that less ST depression during exercise correlates with a higher likelihood of having the condition.
* **Resting Blood Pressure and Target** - There is no clear pattern between resting blood pressure and the target variable, suggesting it may not be a strong predictor of the condition.



**Figure 4 - Correlation matrix.**

* Key indicators for the target condition include a higher max heart rate (negatively correlated at -0.41), the presence of exercise-induced angina (positively correlated at 0.48), a steeper ST slope (positively correlated at 0.51), and certain types of chest pain (positively correlated at 0.46).

# **3. Data Preparation**

## **3.1. Data Selection**

The propose of this project is to identify if heart disease exist or not. From looking on our data we found the 'target' attribute which means our data is pre-labelled (1: heart disease, 0: no heart disease) Therefore, we used 'target' as our target for the prediction of risk for heart disease.

## **3.2. Data Cleaning**

We have 11 features which is not allot and they seem to be pretty important medical risk factors for deciding if a patient will or will not be predicted as at risk for heart disease. we decided that no column will be deleted.

## **3.3. Filling Missing Values**

We don’t have any missing values in our dataset. But another issue we identified is related to the 'cholesterol' attribute. We observed that some rows have a value of '0', which implies that there is no measure for this attribute ('cholesterol' cannot be 0 in real life).

תמונה שמכילה תרשים, צילום מסך, עלילה, קו

התיאור נוצר באופן אוטומטי So, we replaced the value '0' with the mean cholesterol level of non-zero entries to ensure data accuracy. Following this, a log transformation was applied to normalize the distribution of cholesterol levels, making the data more symmetric and reducing the impact of outliers. This transformation enhances the data's suitability for statistical analysis and modelling by stabilizing variance and potentially improving model performance.

**Figure 5 - cholesterol feature after mean and log normalization.**

**Note:** in the appendices you will be able to find the info of the data that shows that there are no missing values.

## **3.4. Data Construction**

We built two custom metric/features that simulates the calculations of risk score for Heart disease in the real world (medicine):

* **Framingham Risk Score (FRS):** estimates the 10-year risk of developing cardiovascular diseases, such as coronary artery disease, stroke, and heart failure.
* **Total Cardiovascular Risk Score (TCRS)**: broader assessment tool that evaluates the overall risk of cardiovascular events by incorporating multiple risk factors and sometimes additional markers like family history and lifestyle factors.

**Table for explaining method calculations:**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Framingham Risk Score (FRS)** | **Total Cardiovascular Risk Score (TCRS)** |
| **Risk Factors Included** | Age, cholesterol, systolic blood pressure, fasting blood sugar | Various risk factors including advanced biomarkers and lifestyle factors |
| **Scoring Method** | **Point-based system**: **Age Points**: Points are assigned based on age brackets, with older ages receiving more points indicating higher risk. **Cholesterol Points**: Points are assigned based on cholesterol levels, with higher cholesterol resulting in more points **BP Points**: Systolic blood pressure is divided into categories, with higher pressures receiving more points. **Fasting Blood Sugar Points**: One point is added if fasting blood sugar is above 120 mg/dL, indicating higher risk. At the end, all the points are summed to calculate the total risk score. | **Weighted sum of normalized and transformed risk factors**: **Log Transformation**: Apply logarithmic transformation to normalize data and reduce skewness. **Normalization**: Use MinMaxScaler to scale values between 0 and 1, ensuring comparability across different features. **Factor Weight Calculation**: Calculate weights based on the correlation of each feature with the target variable, reflecting their importance. **Calculate TCRS**: Compute TCRS for each row by summing the weighted factors for a comprehensive risk score. |
| **Complexity** | Simple calculation using predefined point tables | Complex calculation involving log transformation, normalization, and weighting |
| **Output** | Risk percentage - numeric | Comprehensive risk score - numeric |

**Table 2 - Explanation of new feature construction.**

Using FRS and TCRS as features will the fill the dataset with diverse risk assessments. FRS provides traditional clinical insights, while TCRS offers a comprehensive evaluation including lifestyle and biomarkers. This dual risk score methods may enhances the model's predictive accuracy by integrating the strengths of both scoring systems independently.

**Note:** it’s important to say that they are custom risk score depending on the data we have and probably not accurate like in the literature of medicine.

## **3.5. Data Formatting**

We changed the format of the ST slope feature and observed that it had a high correlation with the target variable (0.51) in the correlation graph. Using the dummy method, we separated the different categories within this feature into four separate binary features. This approach assigns a value of 1 to the occurring category and 0 to the others, helping us understand which specific category has the most influence on the likelihood of heart disease. The conclusion that the variable 2 (flat sloping, Link: [Information about sloping](https://litfl.com/st-segment-ecg-library/) ) has the most influence on it we will show it on the new correlation graph:

תמונה שמכילה טקסט, צילום מסך, צבעוני, דפוס

התיאור נוצר באופן אוטומטי

**Figure 6 - new correlation graph, with new features.**

# **4. Modeling**

## **4.1. Models Used**

From the papers, we can see that most authors used the following models: Support Vector Machine (SVM), Decision Tree, K-Nearest Neighbours (KNN), Naïve Bayes, Logistic Regression, and Random Forest. To enhance our analysis and capture a broader range of model characteristics, we decided to expand our approach to include additional advanced machine learning models that can handle complex patterns, non-linear relationships, and provide good performance even with diverse feature types and potential missing values.

**Models we tested on:**

|  |  |  |
| --- | --- | --- |
| * SVM * Decision Tree * KNN | * Naïve Bayes * Logistic Regression * Random Forest | * XGBoost * Gradient Boosting * Multi-layer Perceptron (MLP) |

In the first running of the models, we used base model form (example: XGBoost()) without cross validation and without hyper parameters. We ran those models with different test sizes that ranges between 10% to 40% to try preventing over fit and under fit and choose the best one depending on the model performance (confusion matrix, metrics). It’s important to note that different models might perform better with different training and test splits, depending on their complexity and the amount of data available. In the appendices you can see the explanation for the models we chose (Table - 7).

## **4.2. Importance Of Features**

On some models we used built-in feature importance for models like Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, and XGBoost because they naturally provide this measure. For Neural Networks, KNN, SVM, and Gaussian Naive Bayes, we employed permutation importance. Permutation importance works by shuffling each feature and measuring the impact on the model’s performance, thereby determining how much the model relies on each feature. it allowed us to evaluate feature relevance across different model types. By identifying the most impactful features, we could define our models better to better build out grid for hyperparameter tuning to try to enhance model performance and generalization.

תמונה שמכילה טקסט, צילום מסך, תרשים, קו

התיאור נוצר באופן אוטומטי

**Figure 7 - Feature importance of each model.**

* ST slope\_1, TCRS, chest pain type, and oldpeak appear as important features across multiple models. Models like Random Forest, Gradient Boosting, and XGBoost assign higher importance to these features because these ensemble methods rely on multiple decision trees, which capture interactions and dependencies among features
* Logistic Regression and Decision Tree highlight features like sex, max heart rate, and cholesterol. Logistic Regression focuses on features with strong linear correlations, while Decision Trees emphasize features that provide the best splits to reduce predictive accuracy early in the tree.

## **4.3. Hyper Parameter Tuning**

For hyper parameter tuning we used optuna library in phyton .in the process we used grid search and cross-validation for top 4 models who performed best in their base form. We ran 300 trails to try to find the best suited and optimized hyper parameters for the different models.

|  |  |
| --- | --- |
| Model | Parameter Grid |
| Decision Tree | max\_depth = trial.suggest\_int('max\_depth', 1, 50)      min\_samples\_split = trial.suggest\_int('min\_samples\_split', 2, 32)      min\_samples\_leaf = trial.suggest\_int('min\_samples\_leaf', 1, 32) |
| Random Forest | n\_estimators = trial.suggest\_int('n\_estimators', 50, 200)      max\_depth = trial.suggest\_categorical('max\_depth', [None] + list(range(10, 21)))      min\_samples\_split = trial.suggest\_int('min\_samples\_split', 2, 10)      min\_samples\_leaf = trial.suggest\_int('min\_samples\_leaf', 1, 4) |
| Gradient Boosting | n\_estimators = trial.suggest\_int('n\_estimators', 100, 1000)      max\_depth = trial.suggest\_int('max\_depth', 1, 50)      learning\_rate = trial.suggest\_loguniform('learning\_rate', 1e-3, 1.0)      min\_samples\_split = trial.suggest\_int('min\_samples\_split', 2, 32)      min\_samples\_leaf = trial.suggest\_int('min\_samples\_leaf', 1, 32)      subsample = trial.suggest\_uniform('subsample', 0.5, 1.0) |
| XGBoost | 'verbosity': 0,          'max\_depth': trial.suggest\_int('max\_depth', 3, 5),          'learning\_rate': trial.suggest\_loguniform('learning\_rate', 0.01, 0.1),          'n\_estimators': trial.suggest\_int('n\_estimators', 100, 200),          'subsample': trial.suggest\_uniform('subsample', 0.8, 1.0),          'colsample\_bytree': trial.suggest\_uniform('colsample\_bytree', 0.8, 1.0),          'gamma': trial.suggest\_loguniform('gamma', 0.1, 0.2), |

**Table 3 - Hyper parameters used for each model.**

**Note:** For optuna you enter variable type and ranges.

# **5. Evaluation**

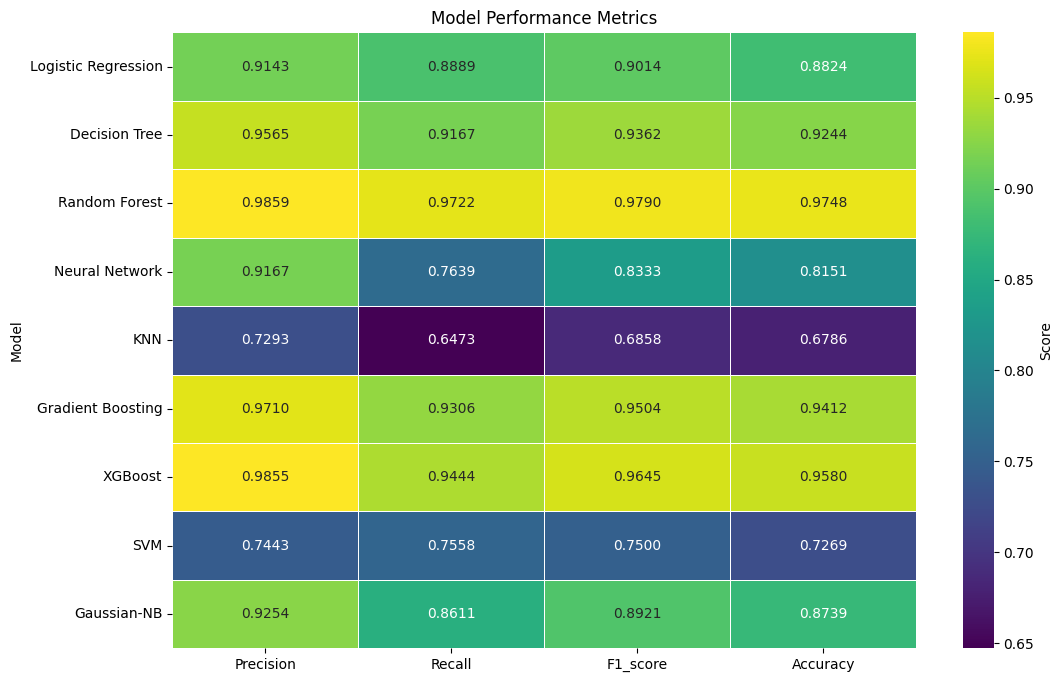
## **5.1. Model Evaluation – Base Models**

comparison of all 9 models using different scoring metrics (based on confusion matrix):

**Explanation of metrics used:**

* **Accuracy:** measures how often the model correctly predicts both positive and negative cases out of all predictions ().
* **Recall:** measures the model's ability to identify all actual positive cases, indicating how well the model detects true positives ().
* **Precision:** measures the accuracy of positive predictions, showing the proportion of true positives among all predicted positives ().
* **F1-Score:** measures how well the model is performing in terms of both detecting all relevant cases and avoiding false alarms ().

**Scores:**



**Figure 8 - evaluating models using different scoring metrics.**

We chose accuracy as the main metric for identifying heart disease risk because it balances true positives and true negative, crucial for reliable diagnostics in the medical world because the cost of mistake can cause fatalities or wrong treatment for patients. Random Forest and XGBoost showed the highest accuracy, making them highly reliable. Gradient Boosting and Decision Tree also performed well, indicating their effectiveness in predicting patient heart disease risk accurately.

## **5.2. Model Evaluation – Hyper Tuned Models**

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Evaluation | Parameters | Improved |
| Decision Tree | **Accuracy: 0.9160**  F1-Score: 0.9275  Recall: 0.8889  Precision: 0.9697 | DecisionTreeClassifier(max\_depth=44, min\_samples\_split=3, min\_samples\_leaf=1) | no |
| Random Forest | **Accuracy:0.9580**  F1-Score: 0.9645  Recall: 0.9444  Precision: 0.9855 | RandomForestClassifier(n\_estimators=76, max\_depth=20, min\_samples\_split=2, min\_samples\_leaf=1) | no |
| Gradient Boosting | **Accuracy: 0.9580**  F1-Score: 0.9645  Recall: 0.9444  Precision: 0.9855 | GradientBoostingClassifier(n\_estimators= 876, max\_depth=43, learning\_rate=  0.009063614378928195, min\_samples\_split=27, min\_samples\_leaf=2, subsample=0.7981349392410265) | yes |
| XGBoost | **Accuracy: 0.9496**  F1-Score: 0.9577  Recall: 0.9444  Precision: 0.9714 | XGBClassifier(colsample\_bytree=0.8214344022679553, gamma=0.13635923084036977, learning\_rate=0.07856097010829145, max\_depth=5, n\_estimators=151, subsample=0.9064226970530631) | no |

**Table 4 - scores after optimization.**

**Note:** in this part after the optimization, we discluded cross validation because it might introduce unnecessary complexity and potentially overfit to validation sets rather than truly testing the model's generalization on unseen data. And in this case for this specific task, we have enough data for not using it so a single train-test split may provide reliable estimate of model performance

Hyperparameter tuning did not improve the model’s performance of the Decision Tree and Random Forest models, even after adjusting parameters like maximum depth, which controls the tree's complexity, and the number of estimators, which impacts robustness. same goes for tuning the XGBoost model with parameters such as learning rate, which influences step size and the number of estimators. In contrast, the Gradient Boosting model showed improved accuracy and other metrics by increasing the maximum depth to capture more complex patterns, adding more estimators to enhance generalization, and adjusting the learning rate to prevent overfitting. Overall, while tuning did not benefit most models, it improved the performance of the Gradient Boosting model.

# **6. Discussion and Conclusions**

In this project, we aimed to predict heart disease using various machine learning models, with Random Forest being the most accurate, achieving an accuracy of 97.48%. Hyperparameter tuning did not improve the accuracy of the Decision Tree, Random Forest, and XGBoost models. In contrast, the Gradient Boosting model showed some improvement after tuning but did not surpass Random Forest's performance. We decided to exclude cross-validation to focus on optimizing the models directly on the training set . In future work, we should research more on the literature of heart disease in the medical area, experiment with more advanced tuning techniques, find additional features that work well with the data, apply better techniques for filling medical data, and add more data to further enhance predictive accuracy.

# **7. Bibliography**

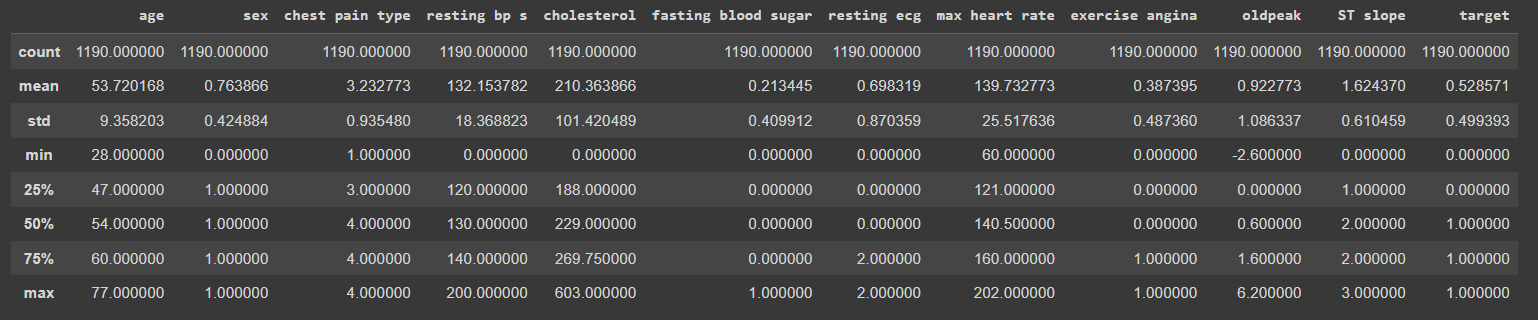
Jindal, H., Agrawal, S., Khera, R., Jain, R., & Nagrath, P. (2021). Heart disease prediction using machine learning algorithms. *IOP Conference Series: Materials Science and Engineering*, *1022*(1), 012072. https://doi.org/10.1088/1757-899X/1022/1/012072

Katarya, R., & Meena, S. K. (2021). Machine Learning Techniques for Heart Disease Prediction: A Comparative Study and Analysis. *Health and Technology*, *11*(1), 87–97. https://doi.org/10.1007/S12553-020-00505-7/TABLES/3

Shah, D., Patel, S., & Bharti, S. K. (2020). Heart Disease Prediction using Machine Learning Techniques. *SN Computer Science*, *1*(6), 1–6. https://doi.org/10.1007/S42979-020-00365-Y/TABLES/3

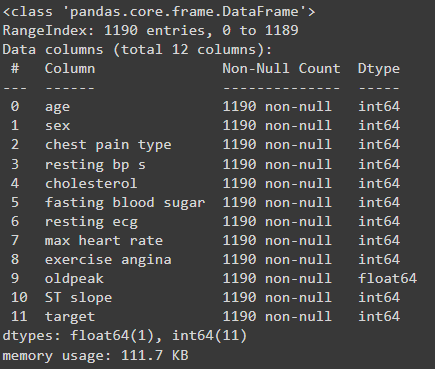
Soni, J., Ansari, U., Sharma, D., of, S. S.-I. J., & 2011‏, undefined. (n.d.). Predictive data mining for medical diagnosis: An overview of heart disease prediction‏. *Academia.Edu‏J Soni, U Ansari, D Sharma, S Soni‏International Journal of Computer Applications, 2011‏•academia.Edu‏*. Retrieved June 15, 2024, from https://www.academia.edu/download/79534142/5a18f6653b56138cd5196d20e2f39de189e3.pdf

# **8. Appendices**

**Descriptive Statistics:**

**Figure 9 - Descriptive Statistics**

**תמונה שמכילה טקסט, גופן, צילום מסך, עיצוב

התיאור נוצר באופן אוטומטיAttribute Description And Missing Values:**

**Figure 10 - Attribute Description and Missing Values**

**Data types and explanation:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1** | age | Age | in years | Numeric |
| **2** | sex | Sex | 1,0 | Binary |
| **3** | chest pain type | chest pain type | 1,2,3,4 | Nominal |
| **4** | resting blood pressure | C | in mm Hg | Numeric |
| **5** | serum cholesterol | cholesterol | in mg/dl | Numeric |
| **6** | fasting blood sugar | fasting blood sugar | 1.0 > 120 mg/dl | Binary |
| **7** | resting electrocardiogram results | resting ecg | 0,1,2 | Nominal |
| **8** | maximum heart rate achieved | max heart rate | 71–202 | Numeric |
| **9** | exercise induced angina | exercise angina | 0,1 | Binary |
| **10** | oldpeak = ST | oldpeak | depression | Numeric |
| **11** | the slope of the peak exercise ST segment | ST slope | 0,1,2 | Nominal |
| **12** | class | target | 0,1 | Binary |

**Table 5 - Data variable explanation part 1.**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| Sex | 1: male, 0: female |
| Chest Pain Type | 1: typical angina,  2: atypical angina,  3: non-anginal pain,  4: asymptomatic |
| Fasting Blood Sugar | fasting blood sugar > 120 mg/dl (1 = true; 0 = false) |
| Resting electrocardiogram results | 0: normal,  1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression > 0.05 mV),  2: showing probable or definite left ventricular hypertrophy by Estes' criteria |
| Exercise induced angina | 1: yes; 0: no |
| The slope of the peak exercise ST segment | 1: upsloping,  2: flat,  3: downsloping |
| Class | 1: heart disease, 0: no disease (normal) |

**Table 6 - Data variable explanation part 2.**

**Model Selection:**

|  |  |  |
| --- | --- | --- |
| **Model** | **Theoretical Approach** | **Reason for Suitability** |
| **Support Vector Machine (SVM)** | SVM seeks to find the optimal hyperplane that separates the data into two classes by maximizing the margin between the classes' closest points (support vectors). It is effective in high-dimensional spaces and robust to overfitting. | Suitable for high-dimensional medical data with clear class separation. |
| **Decision Tree** | Decision Trees split the dataset into subsets based on feature values, resulting in a tree structure where each node represents a decision rule, and each leaf represents an outcome. They are simple to understand and interpret. | Effective for detecting patterns and relationships in medical datasets without requiring assumptions about data distribution. |
| **K-Nearest Neighbors (KNN)** | KNN classifies data points based on the majority class of their nearest neighbours, determined by a distance metric such as Euclidean distance. It is non-parametric and makes no assumptions about the data distribution. | Effective for detecting patterns and relationships in medical datasets without requiring assumptions about data distribution. |
| **Naïve Bayes** | Naïve Bayes classifiers are based on Bayes' Theorem and assume strong (naïve) independence between features. They are simple and efficient, often used in text classification and medical diagnosis. | Efficient and performs well with smaller medical datasets, despite the strong independence assumption. |
| **Logistic Regression** | Logistic Regression models the probability of a binary outcome based on one or more predictor variables using a logistic function. It is easy to implement, interpretable, and provides probabilistic predictions. | Provides clear probabilistic predictions, aiding in understanding the influence of medical risk factors |
| **Random Forest** | Random Forest constructs multiple decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees. It reduces overfitting and provides feature importance measures. | Robust against overfitting and capable of handling complex interactions in medical data. |
| **XGBoost** | XGBoost is an optimized gradient boosting framework that improves performance and speed through parallel processing, regularization, and handling missing data effectively. It is known for its scalability and high predictive accuracy. | Highly efficient and accurate, making it ideal for large and complex medical datasets. |
| **Gradient Boosting** | Gradient Boosting builds models sequentially, each new model attempting to correct errors made by the previous models by minimizing a specified loss function using gradient descent. It is effective in improving model accuracy. | Effective for capturing complex patterns and improving accuracy in medical prediction tasks. |
| **Multi-layer Perceptron (MLP)** | MLP is a type of artificial neural network consisting of multiple layers of nodes, each using a non-linear activation function. MLPs model complex non-linear relationships and are trained using backpropagation to minimize prediction errors. | Capable of modelling complex non-linear relationships in data |

**Table 7 - Models we choose to use in this project and why.**