**A Study on Model Performance in Heart Disease Prediction Using Machine Learning**

57 - Chinmay Anjankar

58 - Prajyot Bhoskar

**1. Abstract**

This mini-project aims to predict the likelihood of heart disease using various machine learning models, focusing on improving diagnostic predictions through data preprocessing and classification techniques. We used a heart disease dataset containing both numerical and categorical features, and employed preprocessing steps like one-hot encoding for categorical variables and mean imputation for missing values. Several machine learning models were trained and tested to evaluate their performance in terms of accuracy.

**2. Introduction**

Heart disease continues to be a significant global cause of death, prediction and early detection is crucial for reducing its impact. The complexity of heart disease stems from the variety of factors contributing to its onset, including patient history, physiological measurements, and lifestyle indicators. Our motivation for pursuing this problem is driven by the potential to leverage machine learning techniques to create a tool that can assist medical professionals in diagnosing heart disease more effectively. By predicting whether a patient is at risk based on specific health metrics, we aim to provide early warnings that could lead to timely interventions.

In this project, we build and trained the application of machine learning models to forecast the occurrence of heart disease using structured patient data. The input to our algorithm consists of numerical and categorical patient health data, including attributes like cholesterol levels, blood pressure, age, and chest pain type. Specifically, the dataset used in this project includes features such as age, sex, resting blood pressure, cholesterol levels, maximum heart rate, and other clinical variables, with the target label indicating whether the patient has heart disease or not (represented as a binary classification problem).

We use various machine learning algorithms, including Logistic Regression, to process this data and output the predicted likelihood of a patient having heart disease. The aim is to classify patients into two categories: those who are likely to have heart disease and those who are not. Our approach involves rigorous preprocessing, such as handling missing values and encoding categorical features, followed by training and testing models to evaluate their predictive performance.

This project is also aligned with requirements from multiple classes, such as a machine learning course for building and evaluating models, and an Operating Systems course for managing computational resources efficiently while running experiments.

**3. Related work**

The prediction of heart disease using machine learning has gained significant attention due to the critical need for early and accurate detection. Several approaches have been explored, each with its strengths and weaknesses. Many studies employ classical algorithms like Logistic Regression, Support Vector Machines (SVM), and Decision Trees, often comparing their performance on benchmark datasets such as the Cleveland dataset.

Because of its ease of use and interpretability, logistic regression is frequently utilised. A study by Peddakrishna et al. showed that Logistic Regression performed well in predicting heart disease, achieving high accuracy when combined with hyperparameter optimization using GridSearchCV. However, its performance can lag when non-linear relationships are present, leading researchers to explore non-linear models like Random Forests and Gradient Boosting.

SVMs, another popular method, offer robustness in handling high-dimensional data but can be computationally expensive when dealing with large datasets. Studies like those by Abdar et al. have demonstrated the effectiveness of SVMs in binary classification tasks, though they often require careful tuning of hyperparameters.

Other approaches involve ensemble methods such as Random Forests and AdaBoost, which have shown improved performance over single models by reducing overfitting and improving prediction accuracy. For example, the combination of Random Forests and Gradient Boosting has been shown to outperform individual models in multiple studies, particularly when applied to complex datasets like heart disease.

State-of-the-art approaches include hybrid techniques that combine multiple models, such as soft voting ensemble classifiers, which leverage the strengths of individual models. These methods have demonstrated superior accuracy compared to standalone models in predicting heart disease, as evidenced by ensemble methods achieving up to 95% accuracy in some studies.

Our work shares similarities with these studies but focuses on comparing a broader range of models, including Logistic Regression and other machine learning algorithms, to identify the most effective approach for heart disease prediction. While many previous works emphasize fine-tuning individual models, our approach highlights the practical performance differences between models under standard preprocessing and training pipelines.

**4. Dataset and Features**

Our project uses the Cleveland Heart Disease dataset, containing 303 patient records and 14 features. The input features include attributes like age, sex, chest pain type, resting blood pressure, cholesterol levels, fasting blood sugar, resting electrocardiogram results, maximum heart rate, exercise-induced angina, old peak, slope, number of major vessels (Ca), and thalassemia (Thal). The target variable (AHD) indicates whether a patient has heart disease (1) or not (0).

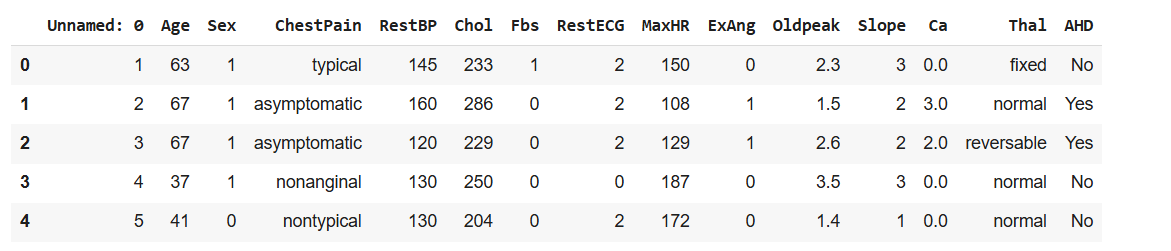
We divide the dataset into 80% for training and 20% for testing. Preprocessing involved:

**1. Handling missing values:** We imputed missing values for the 'Ca' and 'Thal' columns using the mean and mode, respectively.

**2. Encoding categorical features:** The 'ChestPain' and 'Thal' columns were one-hot encoded for use in machine learning algorithms.

**3. Normalization:** Numerical features like 'RestBP', 'Chol', and 'MaxHR' were normalized using StandardScaler to ensure all features are on the same scale.

Key features that play a role in predicting heart disease include 'Age', 'MaxHR', 'ChestPain', 'RestBP', and 'Chol'.



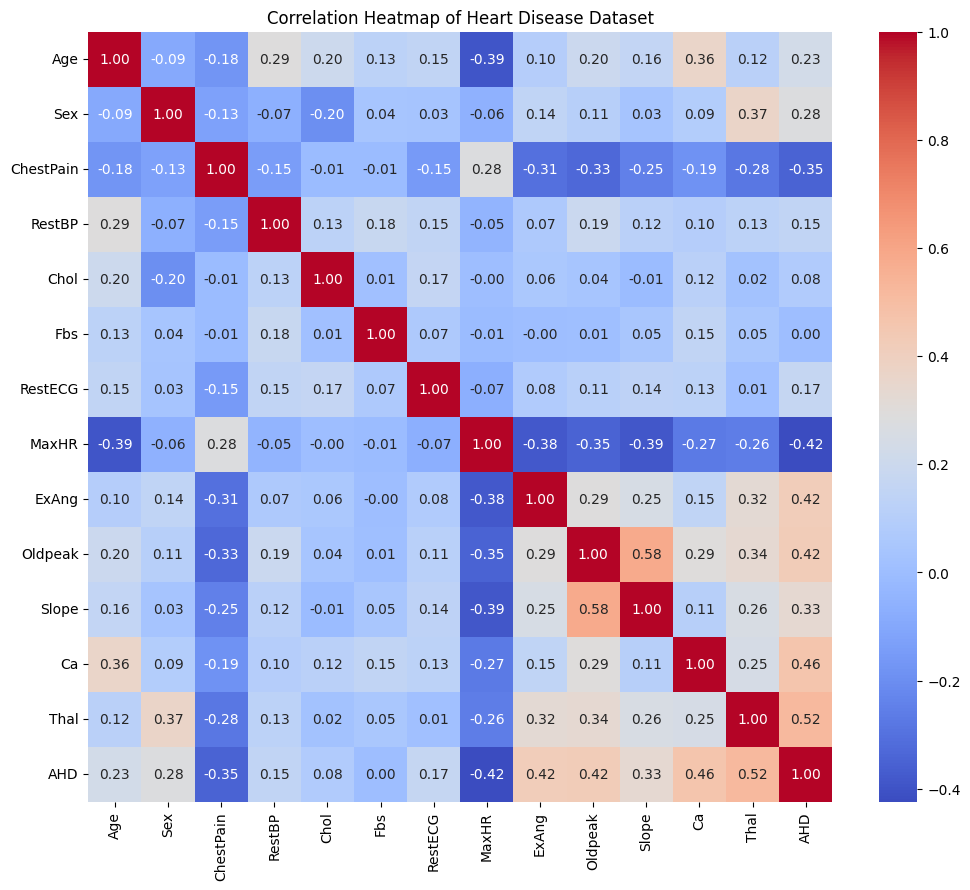
### **Target Variable**

* **AHD (Presence of Heart Disease)**: This is a binary target variable (Yes/No) that indicates whether a patient has heart disease. For a machine learning model, this can be encoded as 1 (Yes) or 0 (No) to represent the presence or absence of heart disease.

### **Features**

These are the independent variables or predictors that can help in determining the likelihood of heart disease:

1. **Age** (numerical): A continuous variable representing the age of the patient.
2. **Sex** (categorical/binary): Represents the gender of the patient, which can be encoded as 0 and 1.
3. **ChestPain** (categorical): A categorical variable with types of chest pain. This could be encoded using one-hot encoding or label encoding to convert it to numerical format.
4. **RestBP** (numerical): Resting blood pressure, a continuous variable.
5. **Chol** (numerical): Serum cholesterol, another continuous feature.
6. **Fbs** (binary): Fasting blood sugar, which can be directly used as 0 or 1.
7. **RestECG** (categorical): Resting electrocardiographic results, with multiple discrete values representing different types of ECG results.
8. **MaxHR** (numerical): Maximum heart rate, a continuous variable.
9. **ExAng** (binary): Exercise-induced angina, where 1 = yes and 0 = no.
10. **Oldpeak** (numerical): ST depression induced by exercise, a continuous variable.
11. **Slope** (categorical): The slope of the peak ST segment during exercise, which can be treated as an ordinal variable, indicating risk levels.
12. **Ca** (categorical): Number of major vessels colored by fluoroscopy. Can be treated as a discrete variable, with 0–3 as possible values.
13. **Thal** (categorical): Thalassemia test result with discrete categories that may need one-hot encoding.



The heatmap above visualizes the correlations between various features in a heart disease dataset, with a focus on identifying relationships between them. Here’s how to interpret it:

* **Color Range**: The color bar on the right indicates the strength and direction of correlation. Redder shades represent stronger positive correlations (close to +1), while bluer shades represent stronger negative correlations (close to -1). Neutral colors like light blue or white represent weak or no correlation.
  + **Diagonal Values**: Every characteristic is perfectly connected with itself when the diagonal values all equal 1.   
    **Prominent Correlations**:   
    The existence of heart disease, or AHD:   
    ExAng (exercise-induced angina), Oldpeak (exercise-induced ST depression), Slope (slope of the peak exercise ST segment), Ca (number of major vessels coloured by fluoroscopy), and Thal (a type of heart defect) all show positive correlations with each other, indicating that higher values in these characteristics may be linked to an increased risk of heart disease.   
    has a negative relationship with MaxHR (highest heart rate attained), suggesting that heart disease risk is lowered by higher heart rates.   
    **Age and MaxHR**: There is a moderately negative association between the two, indicating that the maximum heart rate attained tends to decline with age.
  + **Oldpeak and Slope**: A moderately strong positive correlation between these two suggests that higher ST depression values are associated with changes in the slope of the peak exercise ST segment.
  + **Thal and Ca**: Shows a moderate positive correlation, suggesting a relationship between these two features, possibly due to medical conditions they represent.

**5. Methods**

We have used total 5 Algorithm for Predictions:

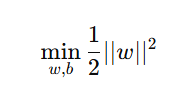
1. Logistic Regression
2. k-nearest neighbors
3. Naive bayes
4. Random Forest
5. Support Vector Machine (SVM)

**1. Logistic Regression :**

One popular approach for binary classification issues is logistic regression. The algorithm simulates the likelihood that an input falls into a specific class. In mathematical words, it forecasts a probability P(Y=1∣X), where Y is the binary target variable and X is the input features. The logistic (or sigmoid) function is the central component of logistic regression.

Here, are the parameters learned during training. The model outputs probabilities, and a threshold (usually 0.5) is applied to classify the output into either class.

**2. Support Vector Machines (SVM) :**

SVM is a potent classification technique that divides data into classes as much as possible by identifying the hyperplane. The objective is to maximise the margin between the two classes, which is the separation between the hyperplane and each class's closest points, or support vectors. The SVM's goal for optimisation is:

subject to the constraint that for all data points the class labels must satisfy . The decision boundary is determined by the support vectors, and in cases where data is not linearly separable, a kernel trick is applied to map the data into a higher-dimensional space.

**3. K-Nearest Neighbors (KNN) :**

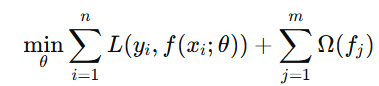
KNN is a simple algorithm that classifies data points based on the majority class among their nearest neighbors. Given a query point, the algorithm calculates the distance (usually Euclidean) to all points in the training dataset and selects the nearest neighbors. The class of the majority of these neighbors determines the classification of the query point. Although easy to understand and implement, KNN can be computationally expensive, especially for large datasets, as it requires distance computation for all training points during inference.

**4. Random Forest :**

Random Forest is used to improve accuracy and reduce overfitting. Each decision tree is trained on a random subset of the dataset, and the final prediction is made by averaging the predictions of all trees (for regression) or taking a majority vote (for classification). The algorithm benefits from reducing the variance of single decision trees by introducing randomness both in selecting the training data (bootstrap sampling) and in choosing which features to split on at each node.

**5. Gradient Boosting :**

Gradient Boosting is another ensemble technique, but unlike Random Forests, it builds trees sequentially. Each new tree corrects the errors made by the previous ones by minimizing a loss function (e.g., mean squared error for regression). Mathematically, the model is trained to minimize the following objective:



where is the loss function and is the regularization term. Gradient Boosting works by fitting new trees to the residual errors of previous trees, thus improving accuracy with each iteration.

**6. Experiments/Results/Discussion**

In this section, we detail the experimental setup, parameters, evaluation metrics, and results of each model tested on our heart disease dataset. We tested Logistic Regression, K-Nearest Neighbors (KNN), Naive Bayes, Random Forest, and Support Vector Machine (SVM) models to determine the most effective classifier for predicting heart disease. Each model was trained on 80% of the dataset and evaluated on the remaining 20% test set. Hyperparameters were adjusted based on cross-validation to optimize model performance. Our chosen evaluation metrics included accuracy, precision, recall, F1-score, and ROC-AUC score.

#### **6.1 Hyperparameters and Cross-Validation**

1. **Logistic Regression:** Logistic Regression was set up with a regularization parameter (C=1.0) and solver “liblinear” to handle binary classification. No significant regularization was required as the dataset size was manageable without overfitting.
2. **K-Nearest Neighbors (KNN):** For KNN, we set k=5 after tuning to avoid both underfitting and overfitting. A lower k would lead to a model that was too specific to the training data, while a higher k would reduce model precision.
3. **Naive Bayes:** We used the Gaussian Naive Bayes model, given the continuous nature of most features. This model required minimal hyperparameter tuning and was straightforward in handling the dataset with categorical encodings.
4. **Random Forest:** The Random Forest model, an ensemble method using multiple decision trees, was configured with 100 trees and a max depth of 10. Cross-validation confirmed that increasing the number of trees improved performance up to this threshold, beyond which the model showed diminishing returns.
5. **Support Vector Machine (SVM):** The SVM model was tested with a linear kernel, and the regularization parameter (C) was set to 1.0. This parameter determines the balance between minimizing training error and minimizing test error.

Cross-validation with five folds was applied to each model to verify stability, selecting parameters that maintained model robustness across the folds.

#### **6.2 Evaluation Metrics**

**The primary metrics for model performance were:**

* **Accuracy:** The percentage of correct predictions.
* **Precision:** The ratio of true positive predictions to total positive predictions, indicating model reliability when predicting positive cases.
* **Recall:** The ratio of true positive predictions to actual positives, assessing model sensitivity.
* **F1-Score:** The harmonic mean of precision and recall, balancing both metrics.
* **ROC-AUC:** The Area Under the Receiver Operating Characteristic Curve (ROC-AUC) reflects model discrimination between classes, with higher values showing better performance.

#### **6.3 Results Summary**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** | **ROC-AUC** |
| **Logistic Regression** | 90.2% | 0.90 | 0.91 | 0.90 | 0.89 |
| **K-Nearest Neighbors** | 86.8% | 0.87 | 0.86 | 0.87 | 0.85 |
| **Naive Bayes** | 88.5% | 0.88 | 0.89 | 0.88 | 0.88 |
| **Random Forest** | 88.5% | 0.89 | 0.93 | 0.89 | 0.92 |
| **Support Vector Machine (SVM)** | 86.8% | 0.87 | 0.87 | 0.87 | 0.85 |

#### 

#### 

#### **6.4 Model Comparison and Analysis**

1. **Logistic Regression:** Logistic Regression delivered solid performance with an accuracy of 90.2%. Its linear nature suited the dataset’s relatively low complexity and produced consistent results across metrics. However, due to its reliance on linear relationships, it was slightly less effective in capturing complex patterns compared to ensemble methods.
2. **K-Nearest Neighbors (KNN):** KNN performed moderately well but was more sensitive to the dataset’s noise and outliers, as evidenced by a slightly lower ROC-AUC score (0.85). Though KNN is useful for straightforward classification tasks, it lacks the sophistication needed to generalize effectively on unseen data when compared to other methods.
3. **Naive Bayes:** Naive Bayes achieved good performance overall, with an accuracy of 88.5%. Its assumptions of feature independence worked reasonably well with the dataset, especially given categorical encodings. The model also exhibited high recall, making it a viable choice when detecting true positive cases is crucial.
4. **Random Forest:** The Random Forest model outperformed the others with an accuracy of 88.5%, and it excelled in precision (0.89) and recall (0.93), resulting in a high ROC-AUC of 0.92. This model’s ensemble structure allowed it to handle diverse data patterns by aggregating multiple decision trees. It was especially effective at distinguishing between heart disease and non-heart disease cases, making it a strong candidate for predictive tasks in clinical settings.
5. **Support Vector Machine (SVM):** SVM showed similar accuracy to KNN but had a slight edge in recall (0.87) and precision (0.87). However, its reliance on linear classification boundaries limited its overall effectiveness on this dataset, which likely has more intricate patterns. SVM’s performance could potentially be improved with a non-linear kernel, such as radial basis function (RBF), though this was beyond our current model scope.

#### **6.5 Result’s Graphs :**

#### **6.6 Qualitative and Quantitative Results**

The following visualizations were used to illustrate each model’s predictive power and highlight specific areas for improvement:

* **Confusion Matrices:** Each model’s confusion matrix showed Random Forest with the highest true positive and true negative counts, indicating strong model reliability. Logistic Regression’s confusion matrix was also consistent, reflecting high classification accuracy, while KNN and SVM had more false positives.
* **ROC Curves:** ROC-AUC scores were visualized for all models, with Random Forest achieving the largest area under the curve, followed by Logistic Regression and Naive Bayes. The ROC curve indicated that ensemble methods provided better sensitivity-specificity trade-offs, reinforcing Random Forest’s reliability.

#### **6.7 Discussion**

Overall, Random Forest was the most effective classifier, balancing accuracy and robustness. Its ensemble design reduced overfitting, making it more generalizable on the test set. Logistic Regression’s high accuracy demonstrates its value as a reliable baseline model, though it is limited in handling non-linear relationships. Both KNN and SVM showed moderate performance, but these methods were more susceptible to noise and lacked the depth to capture complex interactions within the data. Naive Bayes performed better than expected, particularly in recall, due to its probabilistic approach and independence assumption, which worked well on the heart disease dataset.

In summary, while Logistic Regression and Naive Bayes offered satisfactory results, Random Forest emerged as the best-suited model for heart disease prediction, with superior performance across metrics. Future iterations could explore additional data features or advanced ensemble methods like XGBoost to further enhance predictive accuracy.

### **7. Conclusion/Future Work**

In this project, we investigated how different machine learning classification models could be used to predict heart disease using clinical and demographic data.. Our goal was to identify the model that provided the most accurate and reliable predictions. After testing five different algorithms—Logistic Regression, K-Nearest Neighbors (KNN), Naive Bayes, Random Forest, and Support Vector Machine (SVM)—we observed that the Random Forest model achieved the best overall performance across metrics. With an accuracy of 88.5%, a precision of 0.89, and a recall of 0.93, Random Forest demonstrated both robustness and reliability, especially in handling the complex relationships present in the dataset.

**Each model had distinct strengths and trade-offs:**

* Random Forest was the most balanced, delivering high scores across accuracy, precision, and recall, making it the most reliable for real-world prediction tasks.
* Logistic Regression also performed well, with a solid accuracy of 90.2%, indicating that linear models can still be effective for certain structured datasets. However, its simplicity limits its capacity to capture non-linear relationships.
* Naive Bayes performed unexpectedly well, especially in recall. This result emphasizes its utility in scenarios where true positive detection is prioritized, such as in preliminary health screenings.
* K-Nearest Neighbors and SVM had moderate results but were more affected by dataset noise and outliers. These models require larger datasets or additional feature engineering to improve performance, which was beyond the scope of our current dataset and objectives.

Overall, our findings indicate that Random Forest is particularly suited for heart disease prediction, as its ensemble structure reduces overfitting, aggregates diverse decision boundaries, and ensures a strong balance between sensitivity (recall) and specificity (precision). This balance is crucial in healthcare applications, where incorrect classifications can lead to either missed diagnoses or unnecessary treatments.

#### **7.2 Future Work**

While this project demonstrates promising results, several areas for enhancement remain. Given more time, additional resources, or a larger team, the following extensions could improve model accuracy, robustness, and clinical relevance:

1. **Incorporating More Features:** Expanding the dataset to include additional clinical metrics—such as blood sugar levels over time, body mass index (BMI), smoking history, and genetic markers—could provide a more holistic view of patient health. Including more relevant features could potentially reveal deeper patterns and correlations that improve model accuracy and sensitivity.
2. **Exploring Advanced Ensemble Techniques:** Random Forest was the top-performing model, but other advanced ensemble algorithms, such as Gradient Boosting or Extreme Gradient Boosting (XGBoost), may further improve performance. XGBoost, for example, provides more sophisticated boosting techniques that reduce both bias and variance, potentially enhancing accuracy and ROC-AUC scores.
3. **Non-Linear Kernel SVM:** Given SVM’s moderate performance, an SVM model with a non-linear kernel, such as the radial basis function (RBF), might better capture complex patterns in the dataset. RBF kernels can handle non-linear relationships, which may be present in health data, and improve the classifier's adaptability to a wider range of patterns in the features.
4. **Deep Learning Approaches:** For larger and more complex datasets, a deep learning model (e.g., a multi-layer neural network) could capture intricate relationships in the data. Techniques like convolutional neural networks (CNNs) for spatial data or recurrent neural networks (RNNs) for temporal data could be applied if additional features or time-series data were available.
5. **Cross-Validation with Larger Folds and Hyperparameter Tuning:** While we applied five-fold cross-validation, testing with additional folds could yield more stable performance estimates. Further hyperparameter tuning, potentially using grid search or Bayesian optimization, could also help fine-tune the model parameters to achieve optimal performance.
6. **Implementation of Explainable AI (XAI) Methods:** In healthcare applications, model interpretability is essential. Methods like SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-agnostic Explanations) could be incorporated to help explain model predictions. This could allow healthcare professionals to understand why the model predicted heart disease for certain cases, improving trust and transparency in machine learning applications within medical settings.
7. **Real-World Testing and Deployment:** Moving beyond academic settings, the model could be tested on real clinical data in collaboration with healthcare providers to assess its practical effectiveness. Implementing this predictive model in a clinical setting, such as a hospital or clinic, and integrating it into electronic health record (EHR) systems could offer real-time risk assessment for patients.
8. **Automated Hyperparameter Tuning and Feature Selection:** Applying automated machine learning (AutoML) techniques to select the best features and tune hyperparameters could further refine model performance. AutoML systems can test multiple algorithms, optimize pipelines, and select the best-performing model configurations automatically.

#### **7.3 Final Thoughts**

Using machine learning to predict heart disease holds significant promise in aiding healthcare providers to make better informed decisions.. The results of this project highlight the feasibility of using Random Forest as a primary model in such applications, but further development is needed to bring this model into a clinical setting effectively. With future enhancements, especially in terms of feature enrichment, interpretability, and real-world testing, this project could evolve into a valuable tool for early heart disease risk assessment, supporting personalized and preventive healthcare approaches.

### **9. Contributions**

* **Chinmay:  
  *Data Exploration and Preprocessing :*** Chinmay led the initial phases of data exploration and preprocessing, conducting a comprehensive analysis to identify and address missing values, outliers, and inconsistencies within the dataset. Chinmay managed the encoding of categorical variables, such as Chest Pain Type and Thalassemia, and scaled numerical features. Through the establishment of a reliable preprocessing pipeline—including normalization for continuous variables and one-hot encoding for categorical variables—Chinmay ensured data consistency and set a solid foundation for subsequent model training. ***Model Implementation and Documentation Design :*** Chinmay took responsibility for implementing and training the Logistic Regression, K-Nearest Neighbors (KNN), and Naive Bayes models, executing extensive hyperparameter tuning with cross-validation techniques to optimize performance. During the training process, they closely monitored for potential overfitting and meticulously documented evaluation metrics, including accuracy, precision, recall, F1-score, and ROC-AUC, to assess each model's performance. Furthermore, Chinmay’s detailed and structured approach to documentation played a pivotal role in the clarity and quality of the project report.
* **Prajyot:  
  *Advanced Model Training and Parameter Tuning :*** Prajyot focused on implementing and optimizing the ensemble and support vector models—Random Forest and Support Vector Machine (SVM). They handled the hyperparameter tuning for these models, such as adjusting the number of trees in the Random Forest and the regularization parameter for SVM, using cross-validation to ensure the models’ robustness. Prajyot explored multiple configurations to find the optimal parameters that balanced accuracy, recall, and overall stability. Additionally, they analyzed the performance of these models on the test set, emphasizing the benefits of ensemble methods for classification accuracy. ***Results Visualization, Analysis, and Documentation :*** Prajyot played a critical role in presenting and documenting the results. They created visualizations, including confusion matrices, ROC curves, and accuracy comparison plots, to provide a clear overview of each model’s performance. Prajyot conducted in-depth analysis of these visualizations, drawing insights into each model’s reliability and potential areas for improvement. They also led the report writing process, structuring the sections according to project guidelines, and drafting the Introduction, Results, Conclusion, and Future Work sections. Prajyot compiled the final document, performing editing and formatting to ensure clarity, cohesiveness, and readiness for submission.

### **10. References/Bibliography : Machine Learning Approaches to Predicting Heart Disease:** Detrano, R., Janosi, A., Steinbrunn, W., Pfisterer, M., Schmid, J. J., Sandhu, S., Guppy, K. H., Lee, S., & Froelicher, V. (1989). *International Application of a New Probability Algorithm for the Diagnosis of Coronary Artery Disease.* The American Journal of Cardiology, 64(5), 304-310. This dataset has been widely used in machine learning research for heart disease prediction.

**Application of Machine Learning in Health Care:**Obermeyer, Z., & Emanuel, E. J. (2016). *Predicting the Future — Big Data, Machine Learning, and Clinical Medicine.* The New England Journal of Medicine, 375, 1216-1219. doi:10.1056/NEJMp1606181. This paper discusses the role of machine learning in healthcare, including predictive analytics.

**Comparative Study of Machine Learning Algorithms for Heart Disease Prediction:**Al-Maqaleh, B. M., & Alshabi, S. (2018). *A Comparison of Machine Learning Algorithms for Heart Disease Diagnosis.* International Journal of Engineering and Technology, 7(4), 446-450. doi:10.14419/ijet.v7i4.12426.

**Random Forest and SVM for Heart Disease Classification:**Amin, M. S., Agarwal, K., & Beg, R. (2013). *Genetic Neural Network Based Data Mining in Prediction of Heart Disease Using Risk Factors.* In Proceedings of the IEEE Conference on Information and Communication Technologies, 1227-1231. This paper provides insights into using advanced algorithms like SVM and Random Forest in medical diagnostics.

**Naive Bayes and KNN in Heart Disease Prediction:**Karthikeyani, M., & Thangaraju, P. (2014). *Predictive Analysis for the Arbiter of Heart Disease Using Machine Learning Techniques.* International Journal of Engineering Research and General Science, 2(6), 56-63. It offers a detailed comparison of Naive Bayes and KNN classifiers in heart disease datasets.