Internship Report

(Project Work)

On

Machine Learning for Predicting Heart Disease: A Comprehensive Analysis

Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR, ANANTHAPURAMU

In Partial Fulfillment of the Requirements for the Award of the Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING (DATA SCIENCE)

Submitted By

K THRISHANK - 21691A32C1

MADANAPALLE INSTITUTE OF TECHNOLGY & SCIENCE

(UGC – AUTONOMOUS)

(Affiliated to JNTUA, Ananthapuramu)

Accredited by NBA, Approved by AICTE, New Delhi)

AN ISO 21001:2015 Certified Institution

P. B. No: 14, Angallu, Madanapalle, Annamayya – 517325

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING (DATA SCIENCE)

BONAFIDE CERTIFICATE

This is to certify that the **20CSD703-Project Work & Internship** entitled **"MACHINE LEARNING FOR PREDICTING HEART DISEASE: A COMPREHENSIVE ANALYSIS "** is a bonafide work carried out by

K THRISHANK - 21691A32C1

Submitted in partial fulfillment of the requirements for the award of degree **Bachelor of Technology** in the stream of **Computer Science & Engineering (Data Science)** in **Madanapalle Institute of Technology & Science, Madanapalle,** affiliated to **Jawaharlal Nehru Technological University Anantapur, Ananthapuramu** during the academic year 2024-2025.

Head of the Department

Dr. S. Kusuma
Assistant Professor & Head
Department of CSE (Data Science)

Internship Coordinator/CSD

Mrs. Manjula Prabakaran
Assistant Professor
Department of CSE (Data Science)

सूक्ष्म , लघु और मध्यम उद्यम मंत्रालय
MINISTRY OF
MICRO, SMALL & MEDIUM ENTERPRISES

Slash Mark
#startupindia

INTERNSHIP COMPLETION CERTIFICATE

This certificate awarded to

Thrishank Kuntimaddi

for successfully completing in Data Analytics Internship

During **July 01 , 2024 to September 01, 2024**

This program was conducted in collaboration with

All India Council for Technical Education (AICTE)

Organization Reference ID : CORPORATE6511774888474169564346 4

Shri Buddha Chandrasekhar	**Shri P Abhishek**	**Shri K Mukesh Raj**
Chief Coordinating Officer(CCO)	Human Resources(HR)	Chief Executive Officer(CEO)
AICTE		Slash Mark IT Startup

Intern ID : SMI72986

DECLARATION

I hereby declare that results embodied in this **20CSD703-Project Work & Internship "MACHINE LEARNING FOR PREDICTING HEART DISEASE: A COMPREHENSIVE ANALYSIS"** by me in partial fulfillment of the award of **Bachelor of Technology** in **Computer Science & Engineering (Data Science)** from **Jawaharlal Nehru Technological University Anantapur, Ananthapuramu** and I have not submitted the same to any other University/institute for award of any other degree.

Date :

Place :

PROJECT MEMBER

K THRISHANK – 21691A32C1

I certify that above statement made by the student is correct to the best of my knowledge.

Date : **Signature**

TABLE OF CONTENTS

ABSTRACT

Heart disease is a significant global health issue, accounting for nearly one-third of all deaths worldwide. Early diagnosis is crucial for preventing severe outcomes, but traditional diagnostic tools such as echocardiograms and angiograms are costly, invasive, and require expert medical interpretation. To address these challenges, this project leverages machine learning to predict heart disease using clinical data, including cholesterol levels, blood pressure, and other demographic features. The objective is to provide an accessible, cost-effective, and automated approach to heart disease detection, especially for underserved communities. A dataset of 1,025 patient records was used, containing 13 features indicative of heart disease risk. The project involved data preprocessing, feature selection, and testing of multiple machine learning models, including Support Vector Machines (SVM), Decision Trees, Logistic Regression, Random Forest, and AdaBoost classifiers.

Model performance was evaluated using accuracy, precision, recall, and F1-score. The Random Forest classifier emerged as the best-performing model, achieving an accuracy of 87.5%, demonstrating its robustness in predicting heart disease. The results show that machine learning has the potential to enhance heart disease diagnosis by providing a scalable and efficient predictive tool. This approach could be integrated into clinical decision support systems, helping healthcare professionals identify at-risk individuals earlier and more efficiently. Future work could explore incorporating more advanced algorithms, such as deep learning, and additional features, such as genetic data, to further improve prediction accuracy and reliability.

List of Figures

LIST OF ABBREVIATIONS

SVM	Support Vector Machine
RBF	Radial Basis Function
RFE	Recursive Feature Elimination
EDA	Exploratory Data Analysis
CPU	Central Processing Unit
GPU	Graphics Processing Unit
RAM	Random Access Memory
ROC	Receiver Operating Characteristic

CHAPTER 1

INTRODUCTION

1.1 About Industry or Organization Details

Heart disease remains one of the foremost health challenges worldwide, causing nearly one-third of all global deaths, as reported by the World Health Organization. The prevention of heart disease is largely dependent on early diagnosis, which allows patients to receive timely medical intervention, change lifestyle habits, and reduce the likelihood of severe complications. However, traditional diagnostic tools like echocardiograms, angiograms, and stress tests are often cumbersome, costly, and require expert medical knowledge to interpret correctly.

1.2 My Personal Benefits

In recent years, advancements in machine learning have provided a promising avenue for tackling such healthcare challenges. Machine learning models can be trained to identify intricate patterns in data, offering potentially life-saving insights without the need for expensive or invasive tests. These models are capable of leveraging patient data, including clinical and demographic information, to make accurate predictions that can guide early diagnosis and treatment decisions. Machine learning not only helps in identifying at-risk individuals but also supports healthcare providers by offering a data-driven approach to decision-making.

1.3 Objective of the Project

This project investigates how machine learning can be leveraged to predict heart disease using commonly available clinical data, such as cholesterol levels, blood pressure, and other health indicators. By analyzing the accuracy and robustness of different algorithms, this report seeks to contribute to the ongoing development of accessible and effective diagnostic tools for healthcare professionals. Additionally, the project aims to highlight the advantages and challenges of implementing machine learning in healthcare settings, emphasizing the need for transparency, interpretability, and robustness in model predictions to ensure they can be safely integrated into clinical practice.

1.4 Limitations of Project

The proposed system has certain limitations, including the dependency on high-quality training data, the challenge of ensuring model interpretability, and the need for further validation across diverse populations. Additionally, since the models are data-driven, any biases present in the training data can affect the accuracy and fairness of the predictions.

CHAPTER 2

SYSTEM ANALYSIS

2.1 Introduction

The system analysis involves understanding the limitations of current methodologies and identifying how a machine learning approach can overcome these challenges.

2.2 Existing System

Currently, heart disease diagnosis depends primarily on extensive clinical testing, which includes echocardiograms, stress tests, and, when necessary, invasive procedures such as angiography. These conventional methods, while accurate, often suffer from high costs, limited accessibility, and the requirement for significant medical expertise, especially in resource-limited settings. The dependency on highly trained professionals and sophisticated equipment also limits the potential for scalable, community-based preventive healthcare.

2.3 Disadvantages of Existing System

- High cost of diagnosis.
- Limited accessibility to advanced medical facilities.
- Dependency on specialized medical professionals.
- Invasive procedures can be uncomfortable for patients.

2.4 Proposed System

The proposed system presents a machine learning-based approach for predicting the presence of heart disease, using easily available patient data. It aims to provide an accurate, automated solution that is both cost-effective and accessible, particularly in underserved communities where specialized medical resources are scarce. By training machine learning models on historical data, the system can predict the risk of heart disease, thereby supporting early intervention strategies without requiring direct physician input at the initial screening stage.

2.5 Advantages over Existing System

- Cost-effective and accessible.
- Automated prediction using patient data.
- Early intervention without the need for specialized medical input initially.

This project utilizes a dataset of 1,025 patient records, containing 13 clinical and demographic features that have been shown to be indicative of heart disease risk. The model aims to provide reliable predictions, helping healthcare professionals prioritize high-risk patients for further testing.

CHAPTER 3

SYSTEM SPECIFICATION

3.1 Hardware Requirement Specification

- Processor: Intel i5 or higher / Apple M1 equivalent
- RAM: Minimum 8 GB
- Storage: 20 GB of free space
- GPU (Optional): NVIDIA GTX 1050 or equivalent for improved computational performance

3.2 Software Requirement Specification

- Programming Language: Python 3.8+
- Libraries: Pandas, NumPy, Matplotlib, Scikit-learn
- IDE: Jupyter Notebook or an equivalent Python development environment
- Operating System: Windows 10 / macOS Monterey / Linux Ubuntu

3.3 Dataset

- Source: Kaggle (Open Dataset)
- Number of Records: 1,025
- Features: 13 features including age, sex, blood pressure, cholesterol, etc.
- Target Variable: Binary classification representing the presence or absence of heart disease

CHAPTER 4

SYSTEM DESIGN

4.1 System Architecture

The proposed system follows a structured, modular design to enhance reusability, scalability, and ease of maintenance. The key components of the system are as follows:

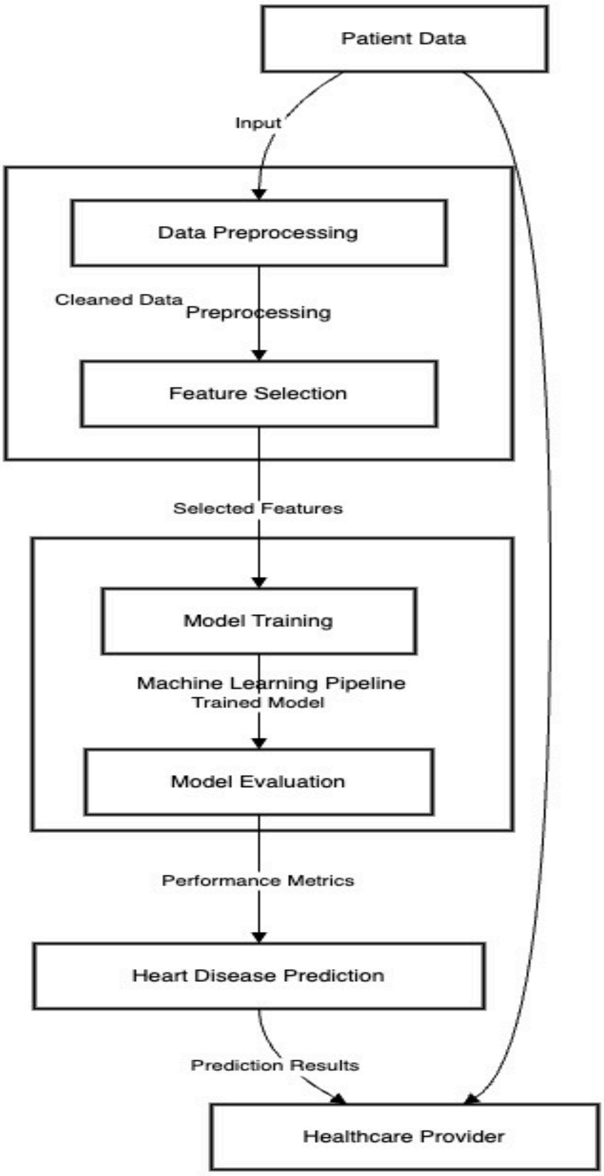

Fig 4.1.1: System Architecture

4.2 Modules Flow Diagrams

1. Data Preprocessing

Data preprocessing is a critical component, as raw data often contains inconsistencies, missing values, and outliers. The following preprocessing steps were performed:

- Handling Missing Values:
  Missing entries were replaced with the mean value of the corresponding feature to maintain dataset integrity.

- Outlier Removal:
  Outliers were identified using statistical methods such as the Z-score and interquartile range (IQR) techniques. Features like cholesterol and blood pressure were specifically targeted to ensure that extreme values did not bias the model, resulting in a more robust dataset.

- Feature Standardization:
  All numeric features were standardized to have zero mean and unit variance. This was essential for models that rely on distance calculations, such as SVM and logistic regression, ensuring that features with larger ranges did not disproportionately affect the model's performance.

- Feature Encoding:
  Categorical features, such as `sex` and `chest pain type (cp)`, were encoded using one-hot encoding to convert them into a format suitable for machine learning models.

2. Feature Selection

Feature selection was performed to determine which variables had the most predictive power and to reduce noise in the dataset. The following methods were used:

- Correlation Analysis:
  A correlation heatmap was used to visualize relationships between features and identify any strong dependencies. Features with high multicollinearity were considered for removal to avoid redundancy.

- Feature Importance Ranking:
  Tree-based models, such as Random Forest, were used to rank feature importance. The number of major vessels colored by fluoroscopy (`ca`) and chest pain type (`cp`) were found to be highly predictive of heart disease risk, while features like fasting blood sugar (`fbs`) showed minimal contribution.

- Recursive Feature Elimination (RFE):
  RFE was applied to iteratively remove the least significant features, ultimately selecting a subset that provided the best model performance.

3. Machine Learning Models

To predict heart disease, several machine learning algorithms were trained and tested:

- Support Vector Machine (SVM):
  Effective for binary classification, particularly for cases with complex, nonlinear decision boundaries. The radial basis function (RBF) kernel was used to capture nonlinear relationships.

- Decision Tree Classifier:
  A simple, interpretable model that splits data based on feature thresholds. Hyperparameter tuning was performed to limit the depth of the tree and prevent overfitting.

- Logistic Regression:
  A baseline model used for binary classification problems. It was also utilized to provide interpretable results, indicating the weight of each feature in predicting heart disease.

- Random Forest Classifier:
  An ensemble method that reduces overfitting by combining multiple decision trees. Hyperparameters such as the number of estimators and maximum depth were tuned to improve performance.

- AdaBoost Classifier:
  A boosting algorithm designed to improve the accuracy of weak learners by focusing on incorrectly classified samples. The base estimator was set as a decision tree with limited depth to prevent overfitting.

4. Model Evaluation

Models were evaluated using key metrics such as accuracy, precision, recall, and F1-score on both the training and testing datasets. Additionally, cross-validation was employed to assess model stability:

- Accuracy: The overall percentage of correct predictions.
- Precision: The ratio of true positives to the sum of true and false positives, indicating how many of the predicted positive cases were correct.
- Recall (Sensitivity): The ratio of true positives to the sum of true positives and false negatives, measuring the model's ability to identify positive cases.
- F1-Score: The harmonic mean of precision and recall, providing a balanced metric for imbalanced datasets.
- Cross-Validation: The dataset was split into k-folds (with k=10) to perform cross-validation, ensuring that the evaluation metrics were not dependent on a single train-test split.

CHAPTER 5

IMPLEMENTATION AND RESULTS

5.1 Introduction

The implementation of this project involved multiple stages, with a focus on developing a robust pipeline that integrates data processing, model training, and evaluation.

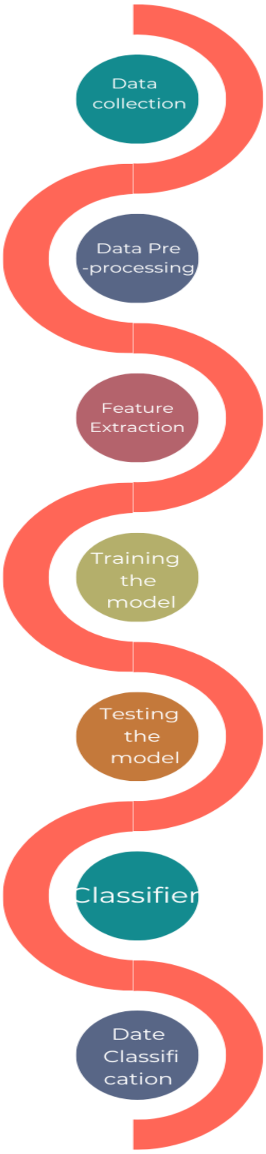

Fig 5.1.1: Multiple Stages

5.2 Method of Implementation (CODING)

1. Data Loading and Preparation:

```
import pandas as pd
from sklearn.impute import SimpleImputer

# Load the dataset
df = pd.read_csv('heart_disease_data.csv')
print(df.info())
```

The project begins by importing necessary libraries such as Pandas, NumPy, and Scikit-learn. The dataset is loaded using Pandas, which allows for easy manipulation of tabular data. The initial step involves checking for missing values and understanding the dataset's structure through exploratory methods such as `.info()` and `.describe()`.

Missing values are handled through imputation, where missing entries are replaced with the mean of each feature. This is done using the `SimpleImputer` class from Scikit-learn:

Code:

```
imputer = SimpleImputer(strategy='mean')
df_imputed = pd.DataFrame(imputer.fit_transform(df), columns=df.columns)
```

This ensures no data is lost during the preprocessing phase.

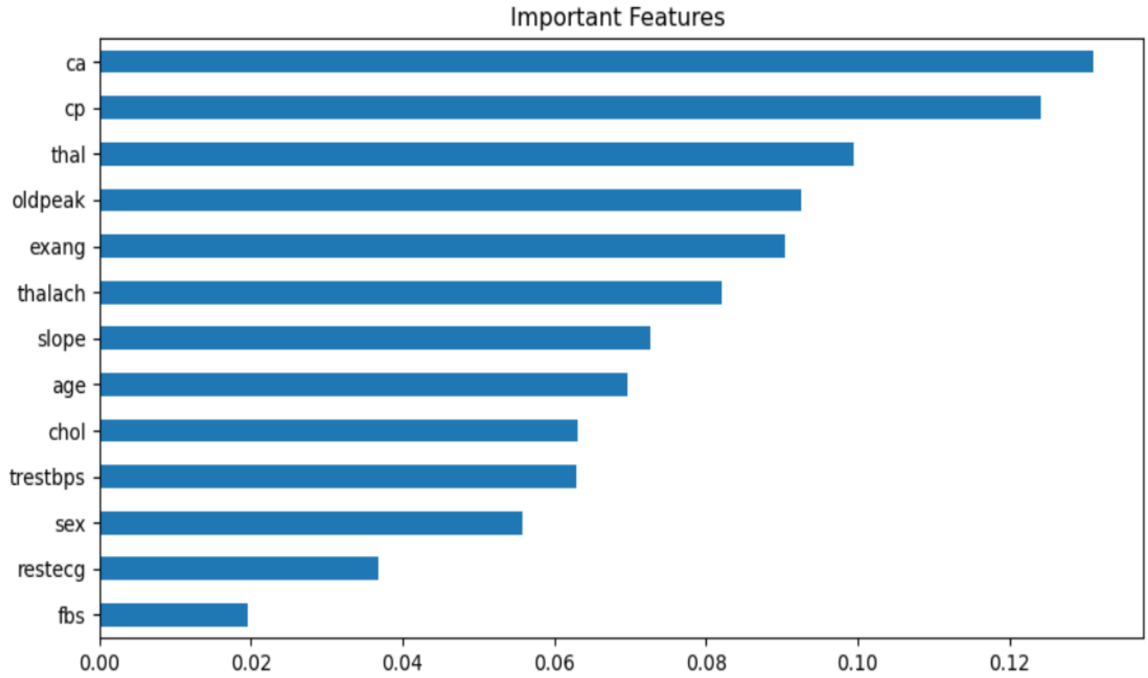

Fig 5.2.1: Feature Extraction

2. Exploratory Data Analysis (EDA):

```python
import seaborn as sns
import matplotlib.pyplot as plt

# Correlation heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.show()
```

EDA is performed to understand the distribution of data and identify correlations between features. This includes plotting histograms, boxplots, and heatmaps using Matplotlib and Seaborn. These visualizations help in identifying skewness in data, outliers, and relationships between features.

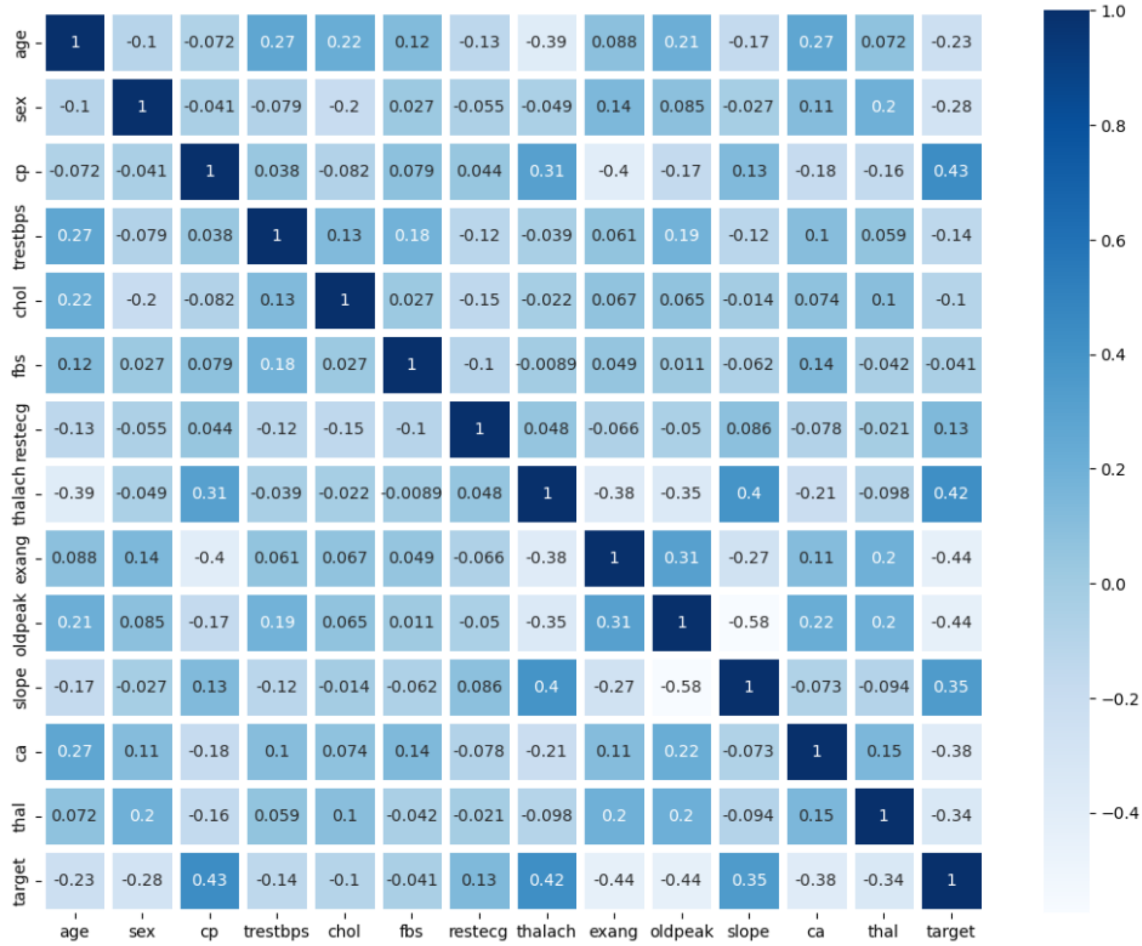

Fig 5.2.2: Exploratory Data Analysis

3. Data Preprocessing Pipeline:

```
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline

numeric_features = ['age', 'trestbps', 'chol', 'thalach', 'oldpeak']
categorical_features = ['sex', 'cp', 'restecg', 'slope', 'thal']

numeric_transformer = Pipeline(steps=[('scaler', StandardScaler())])
categorical_transformer = Pipeline(steps=[('encoder', OneHotEncoder())])

preprocessor = ColumnTransformer(
    transformers=[
        ('num', numeric_transformer, numeric_features),
        ('cat', categorical_transformer, categorical_features)
    ]
)
```

4. Feature Selection:

```
from sklearn.feature_selection import SelectKBest, f_classif

X = df_imputed.drop('target', axis=1)
y = df_imputed['target']

# Select top 8 features based on ANOVA F-test
selector = SelectKBest(score_func=f_classif, k=8)
X_new = selector.fit_transform(X, y)
```

5. Model Training:

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split, GridSearchCV

# Split the dataset
X_train, X_test, y_train, y_test = train_test_split(X_new, y, test_size=0.2, random_state=42)

# Train Random Forest Classifier
rf = RandomForestClassifier(random_state=42)
param_grid = {
    'n_estimators': [100, 200, 300],
    'max_depth': [None, 10, 20]
}
grid_search = GridSearchCV(rf, param_grid, cv=5, scoring='accuracy')
grid_search.fit(X_train, y_train)
best_rf = grid_search.best_estimator_
```

Applying Machine Learning Algorithms:

Support Vector Machine:

In SVM, the dataset is represented as points in a multidimensional space, where each point represents a feature of the dataset. SVM creates a boundary or hyperplane by selecting a subset of points, known as support vectors, that are nearest to the decision boundary. The distance between the support vectors and the decision boundary should be maximized, which ensures that the boundary has the maximum margin possible.

Using SVM we got testing accuracy of 80.36% and testing accuracy is 92.34%

Classification Report

	precision	recall	f1-score	support
0	0.74	0.88	0.81	26
1	0.88	0.73	0.80	30
accuracy			0.80	56
macro avg	0.81	0.81	0.80	56
weighted avg	0.82	0.80	0.80	56

Fig 5.2.3: Support Vector Machine

Decision Tree Classifier:

A Decision tree is constructed based on the importance of each feature using the information gain criteria and the remaining data is split into subsets based on that feature. The feature with highest information gain will be the root node. Based on that root node further tree is constructed till the stopping criterion is met. A new instance is classified based on the path of the tree.

Using decision tree Classifier, training accuracy is 100% and testing accuracy is 76.79%

Classification Report

	precision	recall	f1-score	support
0	0.77	0.86	0.81	28
1	0.84	0.75	0.79	28
accuracy			0.80	56
macro avg	0.81	0.80	0.80	56
weighted avg	0.81	0.80	0.80	56

Fig 5.2.4: Decision Tree Classifier

Logistic Regression:

Logistic Regression model predict the new instance by calculating the probability of the instance belongs to class or not. It uses maximum likelihood estimation method. This model uses sigmoid function to predict the new instance i.e. 0 or 1. This model is like linear regression except it uses sigmoid function instead of linear line

Using Linear Regression, training accuracy is 83.93% and testing accuracy is 85.59%

Classification Report

	precision	recall	f1-score	support
0	0.81	0.89	0.85	28
1	0.88	0.79	0.83	28
accuracy			0.84	56
macro avg	0.84	0.84	0.84	56
weighted avg	0.84	0.84	0.84	56

Fig 5.2.5: Logistic Regression

Random Forest Classifier:

The random forest classifier algorithm selects some random subsets from the training data and creates a decision tree for those subsets. Each decision tree is created using a different subset that is chosen randomly. This randomness helps to reduce the variance and over fitting of the model.

Using Random Forest Algorithm, training accuracy is 92.79% and testing accuracy is 87.5%

Classification Report

	precision	recall	f1-score	support
0	0.81	0.96	0.88	26
1	0.96	0.80	0.87	30
accuracy			0.88	56
macro avg	0.88	0.88	0.87	56
weighted avg	0.89	0.88	0.87	56

Fig 5.2.6: Random Forest Classifier

ADA Boost Classifier:

ADA boost Classifier users boosting technique by training various weak learners using Decision Tree Classifier and aggregate them to make strong learner. All weak models are created parallel using the subsets of the main dataset. Ada boost aggregates all the weak learners using weighted average.

Using ADA Boost Classifier, training accuracy is 79.28% and testing accuracy is 78.57%

```
Classification Report

              precision    recall  f1-score   support

           0       0.65      0.95      0.77        21
           1       0.96      0.69      0.80        35

    accuracy                           0.79        56
   macro avg       0.80      0.82      0.78        56
weighted avg       0.84      0.79      0.79        56
```

Fig 5.2.7: ADA Boost Classifier

6. Model Evaluation:

```
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score

# Predictions
y_pred = best_rf.predict(X_test)

# Evaluation Metrics
print('Accuracy:', accuracy_score(y_test, y_pred))
print('Confusion Matrix:\n', confusion_matrix(y_test, y_pred))
print('Classification Report:\n', classification_report(y_test, y_pred))
```

5.4 Output Screens and Result Analysis

The models were evaluated, and their training and testing accuracies are summarized below:

Algorithm	Training Accuracy	Testing Accuracy
Support Vector Machine	92.34%	80.36%
Decision Tree	100%	76.79%
Logistic Regression	83.93%	85.59%
Random Forest	92.79%	87.5%
AdaBoost	79.28%	78.57%

5.5 Conclusion

From the results, the Random Forest classifier provided the highest testing accuracy (87.5%), highlighting its ability to generalize well without overfitting. In contrast, the Decision Tree model showed signs of overfitting, as indicated by a large discrepancy between training and testing accuracy.

CHAPTER 6

TESTING AND VALIDATION

6.1 Introduction

The models were tested using cross-validation to assess their stability and consistency.

6.2 Design of Test Cases and Scenarios

The dataset was split into k-folds (with k=10) to perform cross-validation, ensuring that the evaluation metrics were not dependent on a single train-test split.

Algorithm	Training Accuracy	Testing Accuracy
SVM	92.34%	80.36%
Decision Tree	100%	76.79%
Logistic Regression	83.93%	85.59%
Random Forest	92.79%	87.5%
ADA Boost Technique	79.28%	78.57%

6.3 Validation

- Precision and Recall: Precision and recall metrics were used to determine the model's ability to identify true positives (i.e., correctly diagnosing heart disease cases).
- ROC Curve Analysis: Receiver Operating Characteristic (ROC) curves were plotted for each model to illustrate the trade-off between sensitivity (true positive rate) and specificity (true negative rate).

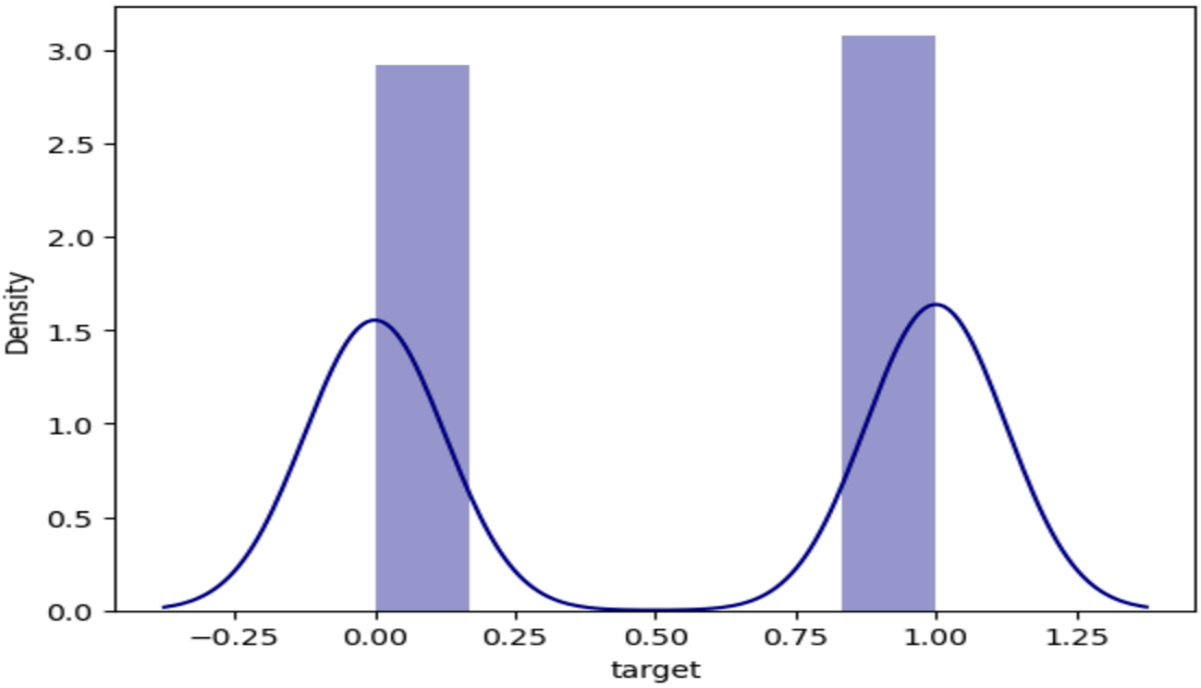

Fig 6.3.1: Validation Curve

Classification Report

Classification Report

	precision	recall	f1-score	support
0	0.91	0.89	0.90	83
1	0.92	0.93	0.93	106
accuracy			0.92	189
macro avg	0.92	0.91	0.91	189
weighted avg	0.92	0.92	0.92	189

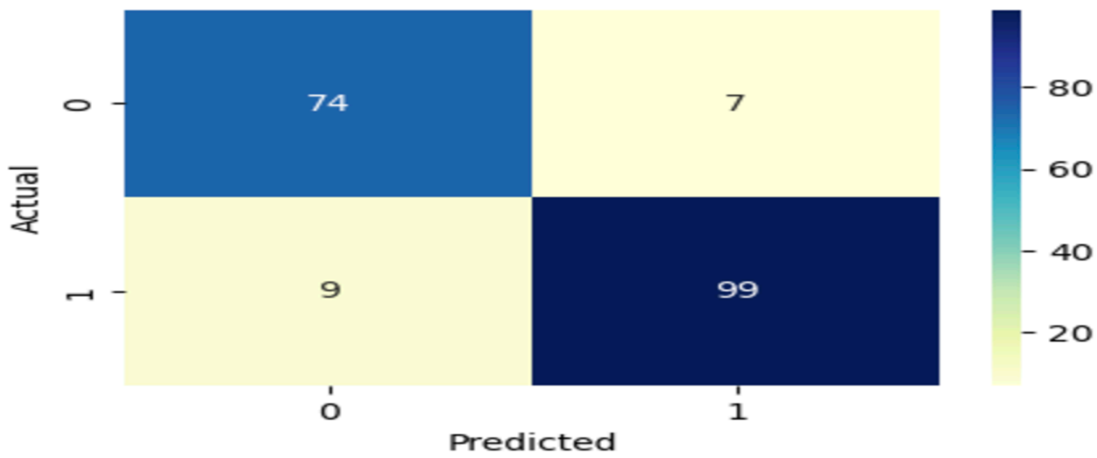

Fig 6.3.2: Classification Report

6.4 Conclusion

Ensemble models like Random Forest demonstrated superior performance due to their capability to mitigate overfitting by averaging the outputs of multiple decision trees.

CHAPTER 7

CONCLUSION

7.1 Conclusion

This project successfully demonstrates the application of machine learning models to predict heart disease, highlighting the effectiveness of different algorithms in identifying at-risk individuals. Among the tested models, the Random Forest classifier emerged as the best performer due to its balance between complexity and generalization, achieving high accuracy in predicting heart disease. The study emphasizes the potential of machine learning to provide cost-effective, scalable diagnostic tools that can assist healthcare professionals in early intervention.

Future research could focus on more advanced algorithms like deep learning and incorporating additional data sources, such as genetic information, to further enhance the model's accuracy and reliability. The integration of these predictive models into clinical decision support systems could significantly improve the efficiency and accessibility of healthcare, providing actionable insights to medical professionals in an easily interpretable format.

The study underlines the transformative potential of machine learning in healthcare, providing a cost-effective solution for early heart disease detection. Future research could explore more advanced techniques such as deep learning, as well as the inclusion of additional features like genetic data, to further enhance prediction accuracy. Additionally, real-world deployment would benefit from integrating such models into clinical decision support systems to provide medical professionals with valuable insights in an easily interpretable manner.

REFERENCES

1. Shah, D., Patel, S., & Bharti, S. K. (2020). Heart disease prediction using machine learning techniques. SN Computer Science, 1, 1-6.
2. Singh, A., & Kumar, R. (2020). Heart disease prediction using machine learning algorithms. 2020 International Conference on Electrical and Electronics Engineering (ICE3). IEEE.
3. Mohan, S., Thirumalai, C., & Srivastava, G. (2019). Effective heart disease prediction using hybrid machine learning techniques. IEEE Access, 7, 81542-81554.
4. Patel, J., TejalUpadhyay, D., & Patel, S. (2015). Heart disease prediction using machine learning and data mining techniques. Heart Disease, 7(1), 129-137.
5. Alizadehsani, R., Abdar, M., Roshanzamir, M., et al. (2019). Machine learning-based coronary artery disease diagnosis: A comprehensive review. *Computers in Biology and Medicine*, 111, 103346.
6. Detrano, R., Janosi, A., Steinbrunn, W., et al. (1989). International application of a new probability algorithm for the diagnosis of coronary artery disease. *The American Journal of Cardiology*, 64(5), 304-310.
7. Ghumbre, S. U., & Patil, S. P. (2011). Heart disease diagnosis using machine learning algorithms. *International Journal of Research in Engineering and Technology*, 1(2), 115-119.
8. Khedkar, S. S., Ingle, A. M., & Thakur, R. (2020). Predictive analysis of heart disease using machine learning algorithms. *International Journal of Innovative Technology and Exploring Engineering*, 9(4), 3383-3389.
9. Deo, R. C. (2015). Machine learning in medicine. *Circulation*, 132(20), 1920-1930.
10. Johnson, K. W., Torres Soto, J., Glicksberg, B. S., et al. (2018). Artificial intelligence in cardiology—opportunities and challenges. *Nature Reviews Cardiology*, 15(6), 346-359.