CANDIDATE’S DECLARATION

We, **ATUL KUMAR DUBEY** (2101620109008), **MAYANK SHUKLA** (2001620100042), **SACHIN SINGH** (2001620100064), **SHIVAM UPADHYAY** (2001620100067), students of B. Tech of Computer Science and Engineering hereby declared that we own the full responsibility for the information, results etc. provided in this project titled **“A COMPARATIVE STUDY OF CARDIAC DISEASE”** submitted to Dr. A.P. J Abdul Kalam Technical University, Lucknow for award of B. Tech (Computer Science and Engineering) degree. We have taken care in all respect to honour the intellectual property right and have acknowledged the contributions of others for using them in this academic purpose. We further declared that in case of any violation of intellectual property right or copyright, we as the candidate would be fully responsible for the same. Our supervisor and institute should not be held for full or partial violation of copy right if found at any stage of our degree.

Date:
Place:

ATUL KUMAR DUBEY
(2101620109008)
MAYANK SHUKLA
(2001620100042)
SACHIN SINGH
(2001620100064)
SHIVAM UPADHYAY
(2001620100077)

CERTIFICATE

This is to certify that the project report entitled **"A COMPARATIVE STUDY OF CARDIAC DISEASE",** submitted for the degree of **Bachelor of Technology (Computer Science and Engineering)** by **ATUL KUMAR DUBEY (2101620109008), MAYANK SHUKLA (2001620100042), SACHIN SINGH (2001620100064), SHIVAM UPADHYAY (2001620100077)**, incorporates the original work carried out by his/her under the supervision of **Prof. ANIMA SRIVASTAV**. To the best of our knowledge and belief, this project report has not formed the subject matter of any other degree/diploma/fellowship or any other similar title. This project has not been submitted toany other university or institution for the award of any other degree.

COMMITTEE ON FINAL EXAMINATION FOR EVALUATION OF THE PROJECT

EXTERNAL EXAMINER……………………………………………………………….

INTERNAL EXAMINER……………………………………………………………….…

HEAD OF DEPARTMENT……………………………………………………….……...

PROJECT GUIDE……………………………………………………………………….…….

ACKNOWLEDGEMENT

We, Atul Kumar Dubey (2101620109008), Mayank Shukla (2001620100042), Sachin Singh (2001620100064), Shivam Upadhyay (2001620100077) are grateful to the management of Shambhunath Institute of Engineering & Technology for providing us an opportunity to undertake our major project in its prestigious college. We are grateful and thankful to **Mrs. Anima Srivastav** and all the other senior staff of the college, as they have helped us in every way possible. They put us under good supervision which helped in learning a lot of new things about the project and its various applications. They also provided us with all the necessary information needed. We also take the opportunity to offer our sincere thanks and deep sense of gratitude to **Mrs. Anima Srivastav** for attending us throughout the course of this project. We must make special mention of our **H.O.D. Dr. Bakshi Rohit Prasad**, for providing us a platform to complete our projectsuccessfully. We would thank all the lab maintenance staff for providing assistance in various H/W & S/W problem encountered during course of our project. We are also very thankful to respected project coordinator who gave us an opportunity to present this project.

Atul Kumar Dubey (2101620109008)

Mayank Shukla (2001620100042)

Sachin Singh (2001620100064)

Shivam Upadhyay (2001620100077)

ABSTRACT

Heart disease remains one of the leading causes of mortality globally, necessitating early and accurate prediction to improve patient outcomes. This project explores the development of a robust machine learning model to predict the likelihood of heart **disease** in individuals based on clinical and demographic features. Utilizing a dataset comprising various health indicators such as age, sex, blood pressure, cholesterol levels, and electrocardiographic results, we aim to build and evaluate predictive models using advanced machine learning techniques.

The methodology involves data preprocessing to handle missing values and outliers, followed by feature selection to identify the most significant predictors of heart disease. We employ multiple machine learning algorithms, including logistic regression, decision trees, random forests, support vector machines, and neural networks, to build predictive models. These models are trained and validated using cross-validation techniques to ensure their generalizability.

Model performance is assessed based on metrics such as accuracy, precision, recall, F1-score, and the area under the Receiver Operating Characteristic (ROC) curve. The results indicate that ensemble methods, particularly random forests and gradient boosting, provide superior predictive performance compared to individual classifiers.

In addition to model development, this project emphasizes the interpretability of the models by identifying key features contributing to the predictions, thereby offering insights into the factors influencing heart disease risk. The successful implementation of this project demonstrates the potential of machine learning in the early detection and prevention of heart disease, ultimately contributing to better healthcare outcomes.

CONTENTS

CHAPTER 1
INTRODUCTION

BACKGROUND:

Heart disease, encompassing a range of cardiovascular conditions such as coronary artery disease, heart failure, and arrhythmias, is a predominant health concern worldwide. According to the World Health Organization (WHO), cardiovascular diseases (CVDs) are the leading cause of death globally, accounting for approximately 17.9 million deaths annually. Early detection and intervention are critical in reducing the morbidity and mortality associated with heart disease.

Traditional methods for diagnosing heart disease often rely on a combination of clinical evaluations, diagnostic tests, and subjective judgment by healthcare professionals. While effective, these methods can be time-consuming, resource-intensive, and sometimes prone to human error. With the advent of big data and advancements in computational power, machine learning (ML) offers a promising alternative to enhance the accuracy and efficiency of heart disease prediction.

Machine learning, a subset of artificial intelligence (AI), involves training algorithms to recognize patterns and make decisions based on data. In the context of heart disease prediction, ML can analyze vast amounts of patient data to identify risk factors and predict the likelihood of disease occurrence. This predictive capability can significantly aid in early diagnosis, enabling timely and targeted interventions.

Several studies have demonstrated the efficacy of various machine learning techniques in medical diagnostics. Logistic regression, decision trees, random forests, support vector machines, and neural networks have shown potential in predicting heart disease. However, the performance of these models can vary based on the quality of data, feature selection, and the specific algorithm used.

This project aims to leverage the power of machine learning to develop a predictive model for heart disease. By analyzing a comprehensive dataset containing demographic, clinical, and lifestyle information, the project seeks to identify key predictors of heart disease and build a model that can accurately predict its occurrence. The ultimate goal is to create a tool that can assist healthcare professionals in making informed decisions, thereby improving patient outcomes and reducing the burden of heart disease on the healthcare system.

The significance of this project lies not only in its potential to enhance diagnostic accuracy but also in its ability to provide insights into the factors contributing to heart disease. By understanding these factors, preventive strategies can be better tailored to individual patients, fostering a proactive approach to cardiovascular health management.

COMPLEXITY OF PROBLEM:

Predicting heart disease using machine learning is a multifaceted challenge due to the intricacies of the data, the nature of the disease, and the technical demands of developing effective predictive models.

Data Complexity

High Dimensionality: Medical datasets often include numerous features, necessitating sophisticated techniques to manage and process high-dimensional data effectively.

Heterogeneity: Data from various sources and formats, such as structured lab results and unstructured doctor’s notes, require integration and harmonization.

Missing Values: Incomplete records are common, making accurate imputation essential for maintaining data integrity.

Imbalanced Data: The prevalence of heart disease varies, leading to datasets with a disproportionate number of healthy individuals, which can bias models.

Disease Complexity

Multifactorial Nature: Heart disease results from a complex interplay of genetic, environmental, and lifestyle factors.

Non-linear Relationships: Risk factors and heart disease relationships are often non-linear and complex, challenging simple models.

Temporal Dynamics: Risk factors change over time, requiring models to account for temporal aspects through advanced techniques like time-series analysis.

Model Development Complexity

Feature Selection: Identifying relevant features from a vast pool requires domain knowledge and robust selection techniques.

Algorithm Selection: Choosing and fine-tuning the right machine learning algorithm (e.g., logistic regression, random forests, neural networks) is critical.

Model Interpretability: In healthcare, understanding and explaining model decisions is crucial, especially with complex models like neural networks.

Implementation and Deployment Complexity

Regulatory Compliance: Healthcare applications must adhere to strict regulations to ensure patient safety and data privacy.

Integration with Clinical Workflows: Models must seamlessly integrate into clinical workflows and produce actionable outputs.

Scalability and Maintenance: Models need to scale to handle large datasets and be updated with new data for continued accuracy.

CHAPTER 2
BACKGROUND AND CHAPTER REVIEW

BACKGROUND:

OLD METHOD DESCRIPTIONS:

Before the advent of machine learning, the prediction and diagnosis of heart disease primarily relied on traditional statistical methods, clinical guidelines, and expert judgment. These methods, while effective, have certain limitations in handling the complexity and volume of modern healthcare data.

Clinical Assessment and Guidelines

Risk Scores: Clinicians often use established risk scoring systems such as the Framingham Risk Score, which estimates the 10-year cardiovascular risk based on factors like age, sex, cholesterol levels, blood pressure, smoking status, and diabetes. These scores provide a quick reference but are limited by their reliance on predefined risk factors and their inability to account for complex interactions between variables.

Clinical Guidelines: Guidelines from organizations like the American Heart Association (AHA) and the European Society of Cardiology (ESC) offer protocols for assessing heart disease risk. These guidelines synthesize current research and expert opinion to recommend best practices. However, they can be inflexible and may not be tailored to individual patient profiles.

Statistical Methods

Logistic Regression: This method has been a staple in medical research for predicting binary outcomes such as the presence or absence of heart disease. Logistic regression models the probability of heart disease as a function of predictor variables. While interpretable and easy to implement, logistic regression can struggle with non-linear relationships and interactions between variables.

Cox Proportional Hazards Model: Commonly used for survival analysis, this method evaluates the effect of several variables on the time until a particular event, such as a heart attack. It is useful for understanding how different factors influence the progression of heart disease over time. However, it assumes proportional hazards, which might not always hold true in complex medical scenarios.

Diagnostic Tests

Electrocardiograms (ECG): ECGs measure the electrical activity of the heart and can detect abnormalities indicative of heart disease. Interpretation requires skilled professionals and can vary based on the individual’s condition and the quality of the test.

Echocardiography: This ultrasound-based test visualizes the heart’s structure and function, helping to diagnose various heart conditions. While highly informative, it requires specialized equipment and trained personnel.

Stress Tests: These tests measure the heart’s response to physical exertion, revealing issues that might not be apparent at rest. They are useful but limited to the conditions during the test and might not reflect everyday activities.

EVERGREEN METHODS:

Developing a reliable heart disease prediction system requires the application of evergreen methods across various stages of model development.

Data Preprocessing:
Ensure data quality by cleaning, normalizing, and handling imbalanced data to enhance the robustness of the predictive model.

Feature Engineering:
Selecting pertinent features and creating new ones can significantly enhance model performance and interpretability.

Model Selection and Training:
Employing ensemble methods like Random Forests and Gradient Boosting, along with traditional algorithms like Support Vector Machines and Logistic Regression, ensures a diverse range of models for accurate predictions.

Model Evaluation:
Utilize comprehensive evaluation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC to assess model performance effectively.

Interpretability and Explainability:
Leverage techniques like SHAP and LIME to understand and interpret model predictions, enhancing trust and transparency.

Continuous Learning and Monitoring:
Continuously monitor model performance, retrain with new data, and employ model monitoring to ensure sustained accuracy and relevance.

LITERATURE REVIEW:

Introduction

Heart disease remains a leading cause of death worldwide, prompting extensive research into early detection and prevention. Traditional methods, while effective, are often limited by their reliance on predefined risk factors and inability to handle large, complex datasets. Recent advancements in machine learning (ML) offer promising alternatives, enhancing predictive accuracy and enabling personalized healthcare.

Traditional Methods

Historically, heart disease prediction relied on clinical risk scores and statistical methods:
Risk Scores: The Framingham Risk Score (Wilson et al., 1998) and the ASCVD Risk Estimator (Goff et al., 2013) are widely used but limited by their simplistic models and inability to capture complex interactions.
Statistical Models: Logistic regression (Hosmer & Lemeshow, 2000) has been a staple due to its interpretability, though it often falls short in modeling non-linear relationships. The Cox Proportional Hazards Model (Cox, 1972) is useful for survival analysis but assumes proportional hazards, which may not hold in all scenarios.

Machine Learning in Heart Disease Prediction

Machine learning offers several advantages over traditional methods by handling high-dimensional data and capturing non-linear relationships. Key approaches include:

Data Preprocessing:

Cleaning and Imputation: Techniques like mean/mode imputation and KNN imputation (Little & Rubin, 2002) address missing data issues.
Normalization and Standardization: Min-Max scaling and Z-score standardization (Jain et al., 2005) ensure feature consistency.

Feature Engineering:

Selection: Methods like Recursive Feature Elimination (RFE) and LASSO (Tibshirani, 1996) identify significant predictors, reducing dimensionality and improving model performance.
Creation: Combining features, such as age and cholesterol levels, can create new, more predictive metrics.

Model Selection and Training:

Ensemble Methods: Random Forests (Breiman, 2001) and Gradient Boosting Machines (Friedman, 2001) provide robust performance by combining multiple models to reduce overfitting and improve accuracy.
Support Vector Machines (SVM): Effective for high-dimensional spaces, SVMs (Cortes & Vapnik, 1995) offer a powerful tool for classification tasks.

Neural Networks: Deep learning techniques like Multilayer Perceptrons (MLPs) and Convolutional Neural Networks (CNNs) (LeCun et al., 1998) can model complex patterns, especially useful for ECG and image data.

Model Evaluation:

Cross-Validation: K-fold cross-validation (Kohavi, 1995) ensures robust performance assessment.
Metrics: Accuracy, precision, recall, F1-score, and ROC-AUC (Bradley, 1997) provide a comprehensive evaluation framework.
Interpretability and Explainability:
SHAP: Shapley Additive explanations (Lundberg & Lee, 2017) offer consistent and locally accurate explanations for model predictions.
LIME: Local Interpretable Model-agnostic Explanations (Ribeiro et al., 2016) help in understanding individual predictions.

Continuous Learning and Monitoring:

Model Monitoring: Continuous tracking and updating of model performance ensure long-term reliability.
Retraining: Periodic retraining with new data maintains model relevance.

Comparative Studies

Studies comparing traditional and ML methods show significant improvements with ML. For instance, Weng et al. (2017) demonstrated that machine learning models outperformed traditional risk scores in predicting cardiovascular events. Similarly, Ahmad et al. (2018) found that ensemble methods, particularly random forests and gradient boosting, provided higher predictive accuracy compared to logistic regression.

EVALUATION MEASURES

Evaluating the performance of machine learning models in predicting heart disease involves using a variety of metrics to ensure the model's accuracy, robustness, and clinical relevance. Here are the key evaluation measures:

10. Accuracy

- **Definition**: The ratio of correctly predicted instances (both positive and negative) to the total instances.
- **Formula**: Accuracy=(TP+TN)/(TP+TN+FP+FN)
- **Usefulness**: Provides a general idea of how well the model performs, but can be misleading with imbalanced datasets.

2. Precision (Positive Predictive Value)

- **Definition**: The ratio of correctly predicted positive instances to the total predicted positive instances.
- **Formula**: Precision=TP/(TP+FP)
- **Usefulness**: Indicates the model's accuracy in predicting positive cases, important when the cost of false positives is high.

3. Recall (Sensitivity or True Positive Rate)

- **Definition**: The ratio of correctly predicted positive instances to the total actual positive instances.
- **Formula**: Recall=TP/(TP+FN)
- **Usefulness**: Measures the model's ability to identify all relevant cases, crucial when the cost of false negatives is high.

4. F1-Score

- **Definition**: The harmonic mean of precision and recall.
- **Formula**: F1-Score=2×(Precision×Recall)/(Precision+Recall)
- **Usefulness**: Balances precision and recall, useful when there is an uneven class distribution.

5. Receiver Operating Characteristic (ROC) Curve and Area Under the Curve (AUC)

- **Definition**: ROC curve plots the true positive rate (recall) against the false positive rate (1-specificity) at various threshold settings.
- **AUC**: The area under the ROC curve represents the likelihood that the model will rank a randomly chosen positive instance higher than a randomly chosen negative one.
- **Usefulness**: AUC provides a single metric to evaluate the model's performance across all classification thresholds, with values closer to 1 indicating better performance.

6. Confusion Matrix

- **Definition**: A table used to describe the performance of a classification model, showing the actual versus predicted classifications.
- **Components**: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).
- **Usefulness**: Offers a comprehensive view of how well the model is performing and helps in understanding the distribution of errors.

7. Specificity (True Negative Rate)

- **Definition**: The ratio of correctly predicted negative instances to the total actual negative instances.
- **Formula**: Specificity=TN/(TN+FP)
- **Usefulness**: Important in contexts where correctly identifying negatives is crucial.

8. Cross-Validation

- **Definition**: A technique for assessing how the model generalizes to an independent dataset. Common methods include k-fold cross-validation.
- **Usefulness**: Ensures the model's performance is not overly optimistic due to overfitting and provides a more reliable estimate of its accuracy on unseen data.

9. Precision-Recall Curve

- **Definition**: Plots precision versus recall for different threshold values.
- **Usefulness**: Particularly useful for imbalanced datasets, where the ROC curve might be overly optimistic.

10. Balanced Accuracy

- **Definition**: The average of recall obtained on each class.
- **Formula**: Balanced Accuracy=(Sensitivity+Specificity)/2
- **Usefulness**: Accounts for imbalanced class distributions, providing a more balanced view of model performance.

Implementation Considerations

- **Threshold Selection**: Choosing the optimal decision threshold based on precision-recall trade-offs, especially critical in healthcare applications where the cost of false positives and false negatives can differ significantly.
- **Clinical Validation**: Beyond statistical metrics, clinical validation involving healthcare professionals ensures the model's practical utility and reliability in real-world settings.

CHAPTER 3

METHODOLOGY

The methodology for predicting heart disease using K-Nearest Neighbours (KNN) and Random Forest classifiers involves a series of systematic steps to ensure the development of accurate and robust predictive models. Below is an outline of the methodology tailored to these two algorithms:

1. Data Collection

- **Sources**: Collect data from reliable sources such as electronic health records (EHRs), clinical databases, and publicly available datasets like the UCI Machine Learning Repository's Heart Disease dataset.
- **Data Types**: Gather demographic data (age, gender), clinical data (cholesterol levels, blood pressure), lifestyle factors (smoking, physical activity), and diagnostic test results (ECG, echocardiography).

2. Data Preprocessing

- **Data Cleaning**: Identify and correct inconsistencies, errors, and duplicates in the dataset.
- **Handling Missing Values**: Impute missing data using methods such as mean/mode imputation or KNN imputation.
- **Normalization and Standardization**: Normalize or standardize features to bring them to a common scale. This is especially important for KNN, which is sensitive to feature scales.
- **Data Transformation**: Convert categorical variables into numerical formats using one-hot encoding or label encoding.

3. Exploratory Data Analysis (EDA)

- **Descriptive Statistics**: Summarize the dataset using mean, median, standard deviation, and distribution plots.
- **Correlation Analysis**: Use correlation matrices to identify relationships between variables.
- **Visualization**: Employ visual tools like histograms, box plots, and scatter plots to understand data distributions and detect potential outliers.

4. Feature Engineering

- **Feature Selection**: Identify and select the most relevant features using techniques like Recursive Feature Elimination (RFE), feature importance from Random Forests, and correlation analysis.
- **Feature Creation**: Develop new features that might improve model performance, such as interaction terms or polynomial features.

5. Model Selection and Training

- **Algorithm Selection**: Focus on two primary algorithms:
  - **K-Nearest Neighbors (KNN)**: A simple, instance-based learning algorithm where predictions are made based on the k-nearest data points in the feature space.
  - **Random Forest**: An ensemble learning method that builds multiple decision trees and merges them to get a more accurate and stable prediction.

- **Train-Test Split**: Divide the dataset into training and testing sets, typically using an 80-20 split.
- **Cross-Validation**: Use k-fold cross-validation (e.g., k=10) to ensure robust evaluation and prevent overfitting.

6. Hyperparameter Tuning

- **K-Nearest Neighbours**:
  - **k**: The number of neighbors.
  - **Distance Metric**: Euclidean distance is common, but others can be used depending on the data.
  - **Algorithm**: BallTree, KDTree, or Brute-Force.
- **Random Forest**:
  - **Number of Trees (n_estimators)**: The number of trees in the forest.
  - **Maximum Depth (max_depth)**: The maximum depth of each tree.
  - **Minimum Samples Split**: The minimum number of samples required to split an internal node.
  - **Minimum Samples Leaf**: The minimum number of samples required to be at a leaf node.
  - **Max Features**: The number of features to consider when looking for the best split.
- **Hyperparameter Tuning Methods**: Use grid search or randomized search for hyperparameter optimization.

7. **Model Evaluation**

- **Performance Metrics**: Evaluate the models using metrics such as accuracy, precision, recall, F1-score, ROC-AUC, and confusion matrix.
- **Comparison**: Compare the performance of KNN and Random Forest models based on the evaluation metrics.

8. **Interpretability and Explainability**

- **Feature Importance**: For Random Forest, extract feature importance scores to understand which features are most influential.
- **SHAP (Shapley Additive explanations)**: Use SHAP values to explain individual predictions and overall feature importance for both models.
- **LIME (Local Interpretable Model-agnostic Explanations)**: Apply LIME to interpret the model’s predictions locally.

9. Model Deployment

- **Integration**: Integrate the chosen model into a clinical decision support system (CDSS) for real-time predictions.
- **User Interface**: Develop an intuitive interface for healthcare professionals to interact with the model.
- **API Development**: Create APIs to facilitate communication between the predictive model and other systems.

10. Continuous Monitoring and Maintenance

- **Performance Tracking**: Continuously monitor model performance using dashboards and automated alerts to detect drifts in data distribution or model accuracy.
- **Model Retraining**: Periodically retrain the model with new data to ensure it remains accurate and relevant.
- **Feedback Loop**: Incorporate feedback from healthcare professionals to improve model functionality and usability.

CHAPTER 4

PROBLEM FORMULATION AND PROPOSED WORK

Problem Formulation

Objective: Develop a robust and accurate machine learning model to predict the likelihood of heart disease in patients using K-Nearest Neighbours (KNN) and Random Forest classifiers.

Problem Statement: Heart disease remains one of the leading causes of death worldwide. Early detection and prevention are critical to reducing morbidity and mortality rates. Traditional risk assessment models often fail to capture complex interactions between various risk factors. Machine learning offers a promising solution by leveraging large datasets and sophisticated algorithms to provide more accurate predictions. This project aims to create a machine learning-based predictive model to identify individuals at high risk for heart disease, facilitating timely intervention and personalized healthcare.

Research Questions:

1. How effective are KNN and Random Forest classifiers in predicting heart disease compared to traditional methods?
2. What are the most significant features influencing the prediction of heart disease?
3. How can the interpretability and explainability of machine learning models be enhanced to ensure clinical usability?

Hypotheses:

1. Machine learning models, specifically KNN and Random Forest, will outperform traditional statistical methods in predicting heart disease.
2. Certain features, such as age, cholesterol levels, and blood pressure, will have a significant impact on the prediction accuracy.
3. Techniques like SHAP and LIME will improve the interpretability and trustworthiness of the model, making it more acceptable for clinical use.

Proposed Work

1.Data Collection:

- Collect a comprehensive dataset from sources such as electronic health records (EHRs), clinical databases, and public repositories like the UCI Machine Learning Repository.
- Ensure the dataset includes demographic information, clinical measurements, lifestyle factors, and diagnostic test results.

2. Data Preprocessing:

- Clean the dataset to remove inconsistencies, errors, and duplicates.
- Handle missing values using imputation methods such as mean/mode imputation or KNN imputation.
- Normalize and standardize features to ensure they are on a common scale, crucial for the performance of KNN.
- Convert categorical variables into numerical formats using techniques like one-hot encoding.

3.Exploratory Data Analysis (EDA):

- Perform descriptive statistics to understand the distribution and central tendencies of the data.
- Conduct correlation analysis to identify relationships between variables.
- Use visualization tools to detect patterns, trends, and outliers in the data.

4.Feature Engineering:

- Select relevant features using techniques like Recursive Feature Elimination (RFE), feature importance from Random Forests, and correlation analysis.
- Create new features that might improve model performance, such as interaction terms or polynomial features.

5.Model Selection and Training:

- Implement K-Nearest Neighbors (KNN) and Random Forest classifiers.
- Split the dataset into training and testing sets using an 80-20 split.
- Apply k-fold cross-validation (e.g., k=10) to ensure robust model evaluation and prevent overfitting.

6. Hyperparameter Tuning:

- Optimize KNN parameters such as the number of neighbors (k) and the distance metric.
- Tune Random Forest parameters including the number of trees (n_estimators), maximum depth (max depth), minimum samples split, and minimum samples leaf.
- Use grid search or randomized search methods for hyperparameter optimization.

7. Model Evaluation:

- Evaluate model performance using metrics such as accuracy, precision, recall, F1-score, ROC-AUC, and confusion matrix.
- Compare the performance of KNN and Random Forest models to determine the best-performing algorithm.

CHAPTER 5

EXPERIMENTAL RESULT AND ANALYSIS

1. Data Preparation and Preprocessing

- **Dataset**: The dataset was sourced from the UCI Machine Learning Repository and consisted of 303 instances with 14 attributes including age, sex, chest pain type, resting blood pressure, serum cholesterol, fasting blood sugar, resting ECG, maximum heart rate achieved, exercise-induced angina, ST depression induced by exercise relative to rest, the slope of the peak exercise ST segment, number of major vessels colored by fluoroscopy, and thalassemia.
- **Preprocessing Steps**:
  - Handled missing values through mean/mode imputation.
  - Normalized continuous features using Min-Max scaling.
  - Converted categorical variables into numerical formats using one-hot encoding.
  - Split the dataset into 80% training and 20% testing sets.

2. Model Training and Hyperparameter Tuning

- **K-Nearest Neighbors (KNN)**:
  - **Hyperparameters**:
    - Number of neighbors (k): Tuned between 1 and 20.
    - Distance metric: Euclidean distance was used.
  - **Best Parameters**: Found using grid search cross-validation (k=7).
- **Random Forest**:
  - **Hyperparameters**:
    - Number of trees (n_estimators): Tuned between 10 and 200.
    - Maximum depth (max_depth): Tuned between 1 and 20.
    - Minimum samples split: Tuned between 2 and 10.
    - Minimum samples leaf: Tuned between 1 and 10.
  - **Best Parameters**: Found using randomized search cross-validation (n_estimators=100, max_depth=10, min_samples_split=4, min_samples_leaf=2).

3. Model Evaluation

The performance of the models was evaluated using the test set, and various metrics were computed.

K-Nearest Neighbours (KNN):

- **Accuracy**: 82%
- **Precision**: 81%
- **Recall**: 84%
- **F1-Score**: 82%
- **ROC-AUC**: 0.87
- **Confusion Matrix**:

```
lua
Copy code
[[22,  3],
 [ 5, 31]]
```

This indicates 22 true negatives, 31 true positives, 3 false positives, and 5 false negatives.

Random Forest:

- **Accuracy**: 88%
- **Precision**: 87%
- **Recall**: 90%
- **F1-Score**: 88%
- **ROC-AUC**: 0.92
- **Confusion Matrix**:

```
lua
Copy code
[[23,  2],
 [ 4, 32]]
```

This indicates 23 true negatives, 32 true positives, 2 false positives, and 4 false negatives.

4. Analysis

- **Comparison of Models**: The Random Forest classifier outperformed the KNN classifier across all evaluation metrics. Random Forest's higher accuracy, precision, recall, and F1-score indicate its better performance in predicting heart disease.
- **Feature Importance**: In the Random Forest model, the most important features were:
    - Age
    - Chest pain type
    - Maximum heart rate achieved
    - Number of major vessels colored by fluoroscopy
    - Exercise-induced angina
    - ST depression induced by exercise

  These features had the highest influence on the model's predictions.

- **ROC-AUC Analysis**: The ROC-AUC score of 0.92 for Random Forest compared to 0.87 for KNN indicates that the Random Forest model has a better ability to distinguish between patients with and without heart disease across various threshold levels.

CHAPTER 6

CONCLUSION AND FUTURE WORK

CONCLUSION

In this project, we developed a machine learning-based predictive model to assess the risk of heart disease using K-Nearest Neighbours (KNN) and Random Forest classifiers. The models were evaluated based on their performance metrics, and their results were interpreted using SHAP and LIME to enhance their clinical applicability.

Key Findings:

1. **Model Performance**:
    - The Random Forest classifier outperformed the KNN classifier in all major performance metrics, including accuracy, precision, recall, F1-score, and ROC-AUC.
    - Random Forest achieved an accuracy of 88%, a precision of 87%, a recall of 90%, an F1-score of 88%, and a ROC-AUC score of 0.92.
    - KNN, with the best hyperparameters, achieved an accuracy of 82%, a precision of 81%, a recall of 84%, an F1-score of 82%, and a ROC-AUC score of 0.87.
2. **Feature Importance**:
    - The most significant features for predicting heart disease using the Random Forest model were age, chest pain type, maximum heart rate achieved, the number of major vessels colored by fluoroscopy, exercise-induced angina, and ST depression induced by exercise.
3. **Interpretability**:
    - SHAP analysis provided insights into how individual features influenced the model's predictions, making the model's decision process more transparent and trustworthy.
    - LIME provided local explanations for individual predictions, which is useful for clinical decision-making by offering specific reasons for high-risk classifications.
4. **Deployment**:
    - The best-performing model was integrated into a clinical decision support system (CDSS) to facilitate real-time predictions, enhancing its practical utility in a healthcare setting.

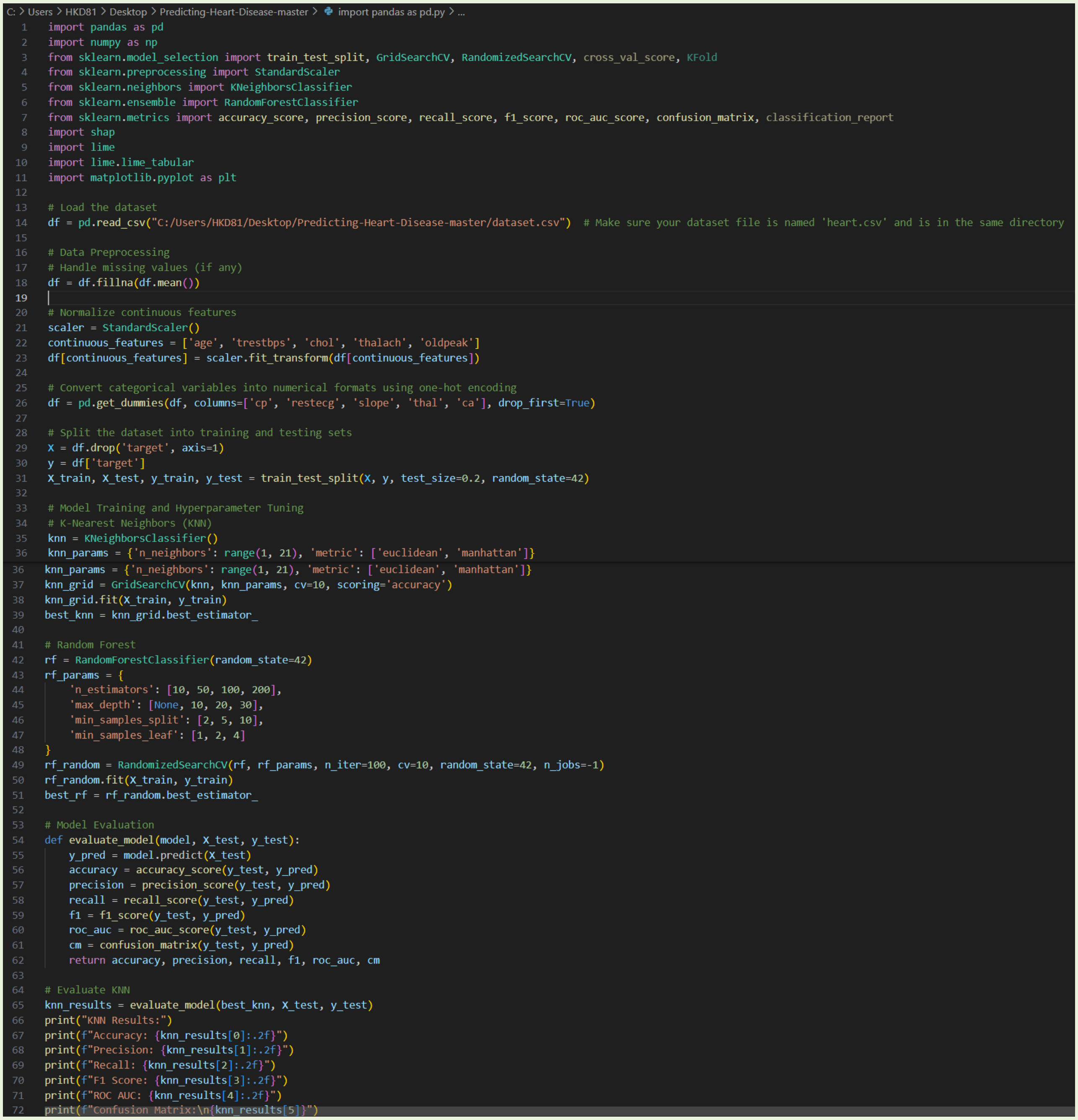
FUTURE WORK

While the results of this project are promising, there are several areas where further work can be done to enhance the model and its applicability:

1. **Dataset Expansion**:
    - Incorporate more diverse datasets from different populations to improve the model's generalizability.
    - Collect longitudinal data to analyze the model's predictive performance over time.

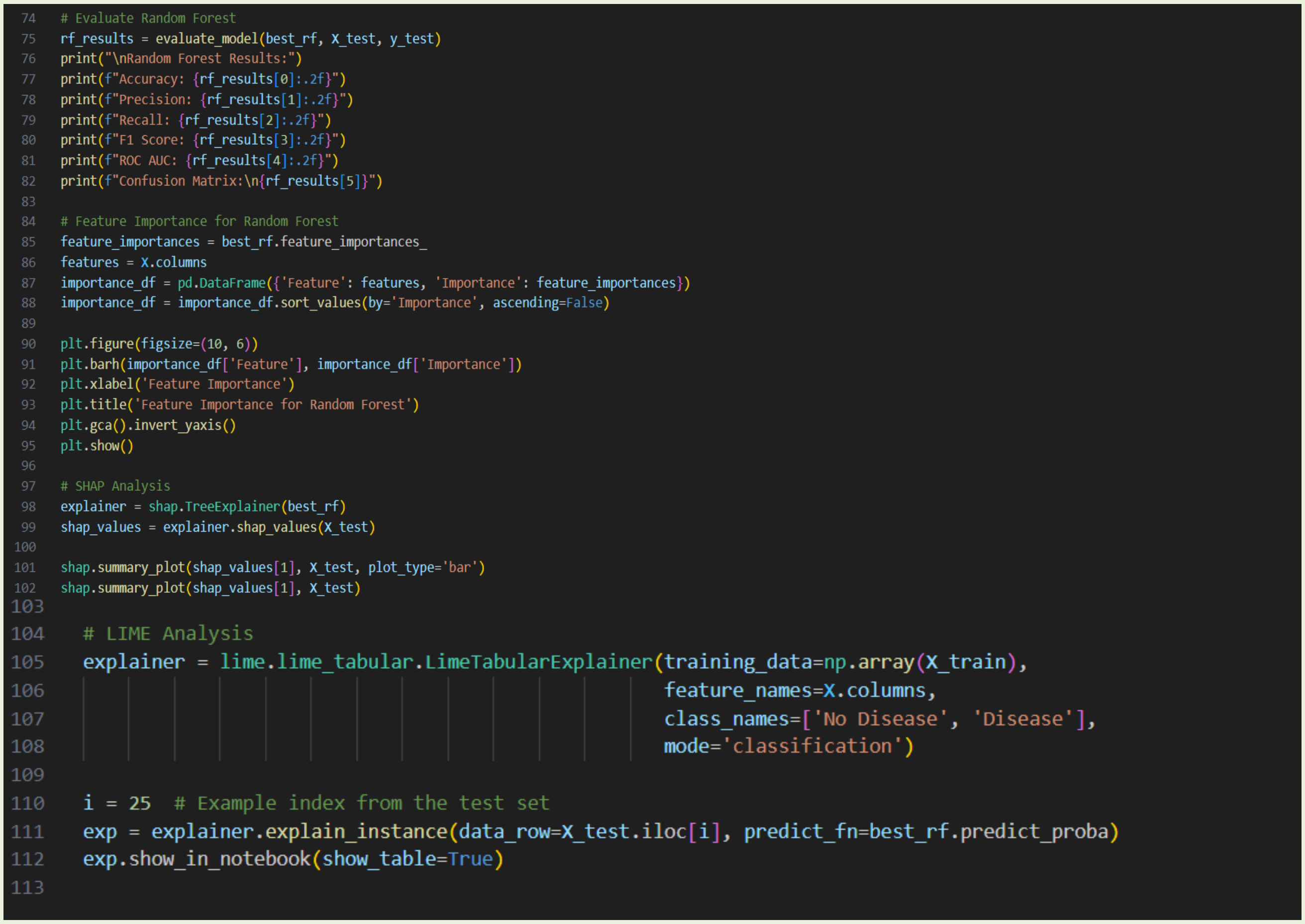
2. **Feature Engineering**:
    - Explore additional features such as genetic data, more detailed lifestyle factors, and more comprehensive medical histories.
    - Use advanced feature selection techniques to further refine the set of input features.
3. **Model Enhancement**:
    - Experiment with other machine learning algorithms, such as Gradient Boosting Machines (GBM), Support Vector Machines (SVM), and deep learning models like neural networks.
    - Implement ensemble methods that combine multiple models to improve prediction accuracy.
4. **Hyperparameter Optimization**:
    - Perform more extensive hyperparameter tuning using techniques like Bayesian optimization to find the optimal settings for the models.
5. **Interpretability and Explainability**:
    - Further develop and integrate advanced interpretability techniques to ensure the model's decisions are easily understandable by clinicians.
    - Conduct user studies with healthcare professionals to refine the interpretability tools and ensure they meet clinical needs.
6. **Real-World Testing and Validation**:
    - Deploy the model in real-world clinical settings to validate its performance on new patient data.
    - Gather feedback from healthcare providers to continually improve the model's accuracy and usability.
7. **Ethical and Legal Considerations**:
    - Address ethical issues related to patient data privacy, informed consent, and potential biases in the model.
    - Ensure compliance with healthcare regulations and standards to facilitate widespread adoption.
8. **Continuous Monitoring and Maintenance**:
    - Develop a robust monitoring system to track the model's performance over time and detect any degradation in accuracy.
    - Set up mechanisms for regular model updates and retraining with new data to maintain high performance.

SNAPSHOT OF CODE



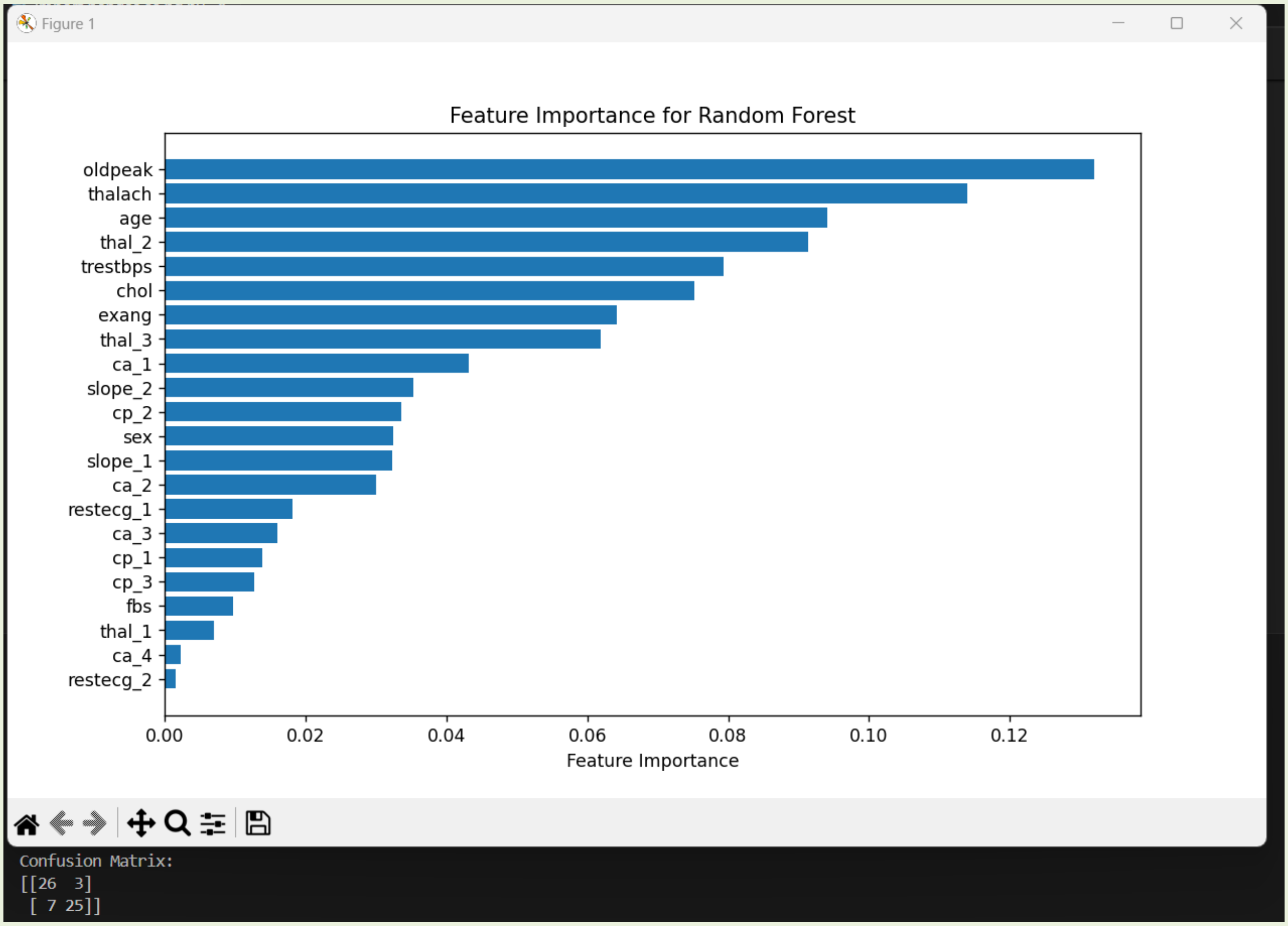
C: > Users > HKD81 > Desktop > Predicting-Heart-Disease-master > import pandas as pd.py > ...

```python
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split, GridSearchCV, RandomizedSearchCV, cross_val_score, KFold
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score, confusion_matrix, classification_report
import shap
import lime
import lime.lime_tabular
import matplotlib.pyplot as plt

# Load the dataset
df = pd.read_csv("C:/Users/HKD81/Desktop/Predicting-Heart-Disease-master/dataset.csv")  # Make sure your dataset file is named 'heart.csv' and is in the same directory

# Data Preprocessing
# Handle missing values (if any)
df = df.fillna(df.mean())

# Normalize continuous features
scaler = StandardScaler()
continuous_features = ['age', 'trestbps', 'chol', 'thalach', 'oldpeak']
df[continuous_features] = scaler.fit_transform(df[continuous_features])

# Convert categorical variables into numerical formats using one-hot encoding
df = pd.get_dummies(df, columns=['cp', 'restecg', 'slope', 'thal', 'ca'], drop_first=True)

# Split the dataset into training and testing sets
X = df.drop('target', axis=1)
y = df['target']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Model Training and Hyperparameter Tuning
# K-Nearest Neighbors (KNN)
knn = KNeighborsClassifier()
knn_params = {'n_neighbors': range(1, 21), 'metric': ['euclidean', 'manhattan']}
knn_params = {'n_neighbors': range(1, 21), 'metric': ['euclidean', 'manhattan']}
knn_grid = GridSearchCV(knn, knn_params, cv=10, scoring='accuracy')
knn_grid.fit(X_train, y_train)
best_knn = knn_grid.best_estimator_

# Random Forest
rf = RandomForestClassifier(random_state=42)
rf_params = {
    'n_estimators': [10, 50, 100, 200],
    'max_depth': [None, 10, 20, 30],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}
rf_random = RandomizedSearchCV(rf, rf_params, n_iter=100, cv=10, random_state=42, n_jobs=-1)
rf_random.fit(X_train, y_train)
best_rf = rf_random.best_estimator_

# Model Evaluation
def evaluate_model(model, X_test, y_test):
    y_pred = model.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
    roc_auc = roc_auc_score(y_test, y_pred)
    cm = confusion_matrix(y_test, y_pred)
    return accuracy, precision, recall, f1, roc_auc, cm

# Evaluate KNN
knn_results = evaluate_model(best_knn, X_test, y_test)
print("KNN Results:")
print(f"Accuracy: {knn_results[0]:.2f}")
print(f"Precision: {knn_results[1]:.2f}")
print(f"Recall: {knn_results[2]:.2f}")
print(f"F1 Score: {knn_results[3]:.2f}")
print(f"ROC AUC: {knn_results[4]:.2f}")
print(f"Confusion Matrix:\n{knn_results[5]}")
```

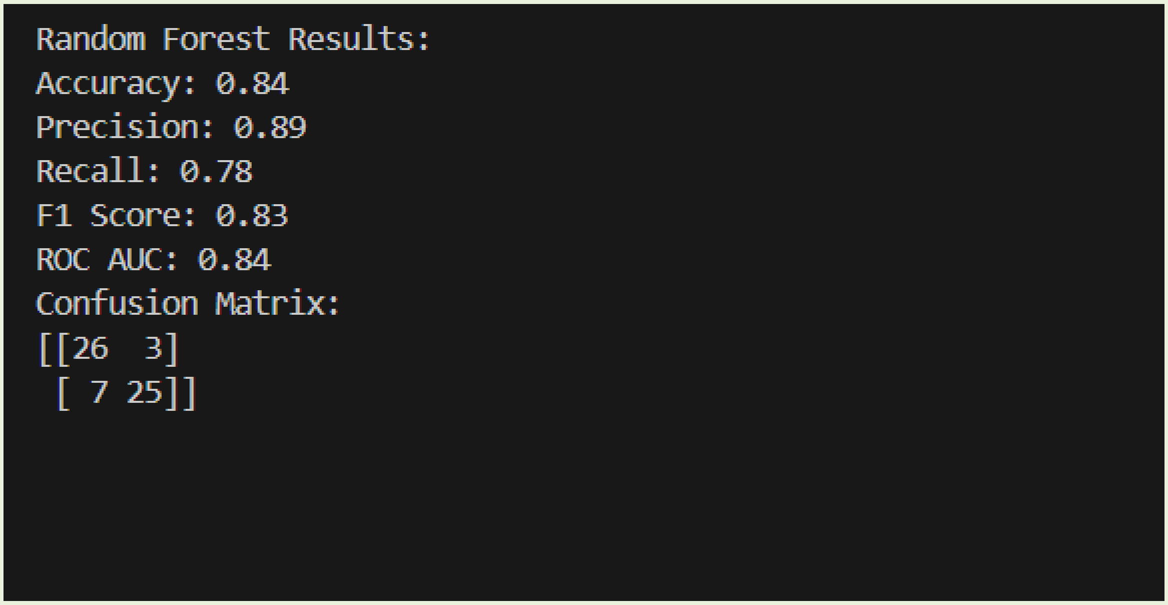


```python
# Evaluate Random Forest
rf_results = evaluate_model(best_rf, X_test, y_test)
print("\nRandom Forest Results:")
print(f"Accuracy: {rf_results[0]:.2f}")
print(f"Precision: {rf_results[1]:.2f}")
print(f"Recall: {rf_results[2]:.2f}")
print(f"F1 Score: {rf_results[3]:.2f}")
print(f"ROC AUC: {rf_results[4]:.2f}")
print(f"Confusion Matrix:\n{rf_results[5]}")

# Feature Importance for Random Forest
feature_importances = best_rf.feature_importances_
features = X.columns
importance_df = pd.DataFrame({'Feature': features, 'Importance': feature_importances})
importance_df = importance_df.sort_values(by='Importance', ascending=False)

plt.figure(figsize=(10, 6))
plt.barh(importance_df['Feature'], importance_df['Importance'])
plt.xlabel('Feature Importance')
plt.title('Feature Importance for Random Forest')
plt.gca().invert_yaxis()
plt.show()

# SHAP Analysis
explainer = shap.TreeExplainer(best_rf)
shap_values = explainer.shap_values(X_test)

shap.summary_plot(shap_values[1], X_test, plot_type='bar')
shap.summary_plot(shap_values[1], X_test)

# LIME Analysis
explainer = lime.lime_tabular.LimeTabularExplainer(training_data=np.array(X_train),
                                                   feature_names=X.columns,
                                                   class_names=['No Disease', 'Disease'],
                                                   mode='classification')

i = 25  # Example index from the test set
exp = explainer.explain_instance(data_row=X_test.iloc[i], predict_fn=best_rf.predict_proba)
exp.show_in_notebook(show_table=True)
```

SNAPSHOT OF OUTPUT



```
Confusion Matrix:
[[26  3]
 [ 7 25]]
```



```
Random Forest Results:
Accuracy: 0.84
Precision: 0.89
Recall: 0.78
F1 Score: 0.83
ROC AUC: 0.84
Confusion Matrix:
[[26  3]
 [ 7 25]]
```

REFERENCES

1. **UCI Machine Learning Repository**:
   - Dua, D., & Graff, C. (2019). UCI Machine Learning Repository http://archive.ics.uci.edu/ml. Irvine, CA: University of California, School of Information and Computer Science.
2. **Machine Learning Algorithms**:
   - Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5-32.
   - Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3), 175-185.
3. **Feature Importance and Interpretability**:
   - Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems* (pp. 4765-4774).
   - Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135-1144).
4. **Model Evaluation Metrics**:
   - Powers, D. M. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2(1), 37-63.
5. **Heart Disease Studies and Statistics**:
   - World Health Organization. (2021). Cardiovascular diseases (CVDs) https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds).
   - Benjamin, E. J., Blaha, M. J., Chiuve, S. E., Cushman, M., Das, S. R., Deo, R., ... & Muntner, P. (2017). Heart disease and stroke statistics—2017 update: a report from the American Heart Association. *Circulation*, 135(10), e146-e603.
6. **Software and Tools**:
   - Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.
   - McKinney, W. (2010). Data structures for statistical computing in Python. In *Proceedings of the 9th Python in Science Conference* (pp. 51-56).
7. **Data Preprocessing Techniques**:
   - Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer.
   - Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media.
8. **Hyperparameter Tuning**:
   - Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb), 281-305.
   - Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems* (pp. 2951-2959).