**Final Project Report: Heart Disease Prediction**

**1. Introduction**

The objective of this project was to develop and deploy a machine learning model capable of predicting the risk of heart disease based on patient health data. The analysis was performed on a dataset containing 303 patient records and 14 features, including age, cholesterol levels, blood pressure, and various clinical measurements. The final output is a simple web application where users can input patient information and receive a prediction.

**2. Data Preprocessing**

Data preprocessing was a critical initial step to prepare the dataset for machine learning. The following steps were taken:

* **Handling Missing Values:** The dataset was found to have no missing values, so no imputation or deletion was necessary.
* **Categorical Encoding:** Features like chest pain type (cp), resting electrocardiographic results (restecg), and thalassaemia (thal) were nominal categorical variables. These were converted into a numerical format using one-hot encoding, creating new binary columns for each category.
* **Feature Scaling:** To ensure that features with larger numerical values did not disproportionately influence the models, numerical features such as age, cholesterol, and blood pressure were standardized using StandardScaler. This process transformed the data to have a mean of 0 and a standard deviation of 1.

**3. Exploratory Data Analysis (EDA)**

Exploratory Data Analysis provided key insights into the dataset's characteristics.

* **Numerical Features:** Histograms revealed the distribution of numerical features, showing that some, like age, followed a relatively normal distribution, while others, like oldpeak, were skewed. Boxplots were used to visualize these distributions and identify outliers.
* **Correlation Analysis:** A correlation heatmap showed relationships between features. For instance, a strong negative correlation was observed between thalach (maximum heart rate) and oldpeak. The target variable showed a positive correlation with features like thalach and chest pain type, and a negative correlation with exang (exercise-induced angina).
* **Target Variable Distribution:** The target variable was found to be nearly balanced (138 for no heart disease, 165 for heart disease), which is ideal as it prevents models from being biased toward the majority class.

**4. Model Development**

The project evaluated four common machine learning classifiers to determine the best-performing model. The data was split into a 70% training set and a 30% testing set.

* **Models Trained:** Logistic Regression, Decision Tree Classifier, Random Forest Classifier, and Support Vector Machine (SVM) were trained on the preprocessed data.
* **Evaluation Metrics:** Model performance was assessed using a comprehensive set of metrics, including:
  + **Classification Report:** Providing precision, recall, and F1-score for each class.
  + **ROC Curve and AUC:** Measuring the models' ability to distinguish between positive and negative classes.
  + **5-fold Cross-Validation:** Providing a robust measure of performance and helping to detect overfitting.
* **Hyperparameter Tuning:** GridSearchCV was applied to each model to find the optimal hyperparameters. This process, for example, determined the ideal max\_depth for the Decision Tree and the best n\_estimators for the Random Forest.
* **Best Model Selection:** After tuning, the **Random Forest Classifier** demonstrated the highest overall performance across all metrics (accuracy, F1-score, and AUC), confirming its robustness and effectiveness for this dataset.

**5. Deployment**

The final, optimized model was deployed as a simple web application using **Streamlit**.

* **Saving the Model:** The best-performing RandomForestClassifier and the StandardScaler object were saved to disk using joblib. This allows the web application to load the trained model and preprocessing steps without retraining.
* **Web Interface:** A Streamlit application was developed to create an interactive and user-friendly interface. It allows a user to input new patient data through forms and select boxes.
* **Prediction:** The app loads the saved model, preprocesses the user's input in the same way the training data was handled (including scaling), and then uses the model to generate a real-time prediction, along with the prediction confidence.

**6. Conclusion**

This project successfully demonstrated a complete machine learning workflow, from initial data exploration and model training to final deployment. The chosen Random Forest model proved to be highly effective for this classification task.

Future improvements could include:

* **More Data:** The model's performance could be further enhanced by incorporating a larger and more diverse dataset.
* **Advanced Algorithms:** Exploring more complex algorithms like XGBoost, LightGBM, or neural networks could yield better results.
* **More Features:** Incorporating additional health data or patient history could provide the model with a richer context for prediction.
* **Explainability:** Implementing tools like SHAP or LIME to explain model predictions could make the application more useful for medical professionals.