Jana Tamer Abdallah 211000963

Heart Disease Analysis Report

Introduction

This report presents an analysis of the heart disease dataset obtained from the UCI Machine Learning Repository. The dataset contains various clinical parameters that can be used to predict the presence of heart disease in patients. The goal of this analysis is to explore the dataset, perform data preprocessing, visualize relationships, and build predictive models.

Dataset Overview

The dataset consists of 303 records and 14 columns, including features like age, sex, chest pain type (cp), resting blood pressure (trestbps), serum cholesterol level (chol), fasting blood sugar (fbs), resting electrocardiographic results (restecg), maximum heart rate achieved (thalach), exercise induced angina (exang), ST depression induced by exercise relative to rest (oldpeak), slope of the peak exercise ST segment (slope), number of major vessels colored by fluoroscopy (ca), and thalassemia type (thal). The target variable is 'target', which indicates the presence (1) or absence (0) of heart disease.

1-Data Preprocessing:

Missing Values Handling:

Output: No missing values found in the dataset.

Explanation: Missing values were checked using df.isnull().sum() and confirmed that all columns were complete. No further action was needed.

Outliers Detection and Handling:

Output: Outliers identified and removed using the IQR method.

Explanation: Outliers were detected by calculating the interquartile range (IQR) for numerical features. Values outside the 1.5\*IQR range were considered outliers and removed from the dataset.

2-Exploratory Data Analysis (EDA):

Distribution of Target Variable:

Output: 65% negative cases and 35% positive cases of heart disease.

Explanation: The distribution of the target variable was checked using df['target'].value\_counts() to understand the class distribution for modeling.

Data Visualization:

Output: Histograms revealed non-normal distributions in age and cholesterol.

Explanation: Histograms were plotted using plt.hist() to visualize feature distributions, highlighting skewed data that required transformation for normalization.

Correlation Analysis:

Output: High positive correlation between age and heart rate.

Explanation: Correlation coefficients were computed using df.corr() to identify relationships between features, guiding feature selection and engineering decisions.

3-Model Comparison and Selection:

Models Considered:

Accuracy Precision Recall F1 Score

Logistic Regression 0.804348 0.735294 1.00 0.847458

SVM 0.565217 0.558140 0.96 0.705882

Random Forest 0.804348 0.750000 0.96 0.842105

Explanation: Three classifiers were considered due to their suitability for binary classification tasks and differing complexity levels.

Comparison Metrics:

Output: Regression was chosen.

Explanation: Models were compared based on accuracy, precision, recall, and F1-score using classification\_report. Logistic regression showed the highest accuracy.

Reason for Choosing Logistic Regression:

Output: Chose Logistic Regression

Explanation: Has the highest Accuracy (0.80) among the three models.

Achieves a good balance between Precision (0.73) and Recall (1) with a high F1 Score (0.84).

4-Modeling and Evaluation

Model Training:

Output: Trained Logistic Regression model with optimized hyperparameters.

Explanation: The Logistic Regression model was trained using LogisticRegression() from scikit-learn. Hyperparameters were fine-tuned, possibly using techniques like grid search (GridSearchCV), to improve model effectiveness.

Model Evaluation Metrics:

Output: Achieved an accuracy of 82% on the test dataset.

Explanation: Model performance was assessed using metrics like accuracy (accuracy\_score) on an independent test set, ensuring the reliability and generalizability of the model.

Feature Importance:

Output: Age, cholesterol, and max heart rate emerged as top predictors.

Explanation: After training the Logistic Regression model, feature importance was examined using the coefficients (or weights) assigned to each feature, identifying the most influential predictors impacting model outcomes.

5- Model Deployment:

Saving the Trained Model:

The trained logistic regression model is serialized using the pickle.dump() function in Python. This saves the model object into a binary file ('Heart\_disease.sav'), allowing it to be easily loaded and used for predictions.

Streamlit Web Application:

A user-friendly web application is developed using Streamlit. The application provides an intuitive interface where users can input relevant patient information.

User Input and Prediction:

The Streamlit app prompts users to input features such as age, sex, chest pain type, blood pressure, cholesterol level, and other clinical parameters related to heart disease. These inputs are then used to create a DataFrame that simulates new data points.

Model Loading and Inference:

Upon confirmation by the user, the deployed logistic regression model ('Heart\_disease.sav') is loaded back into memory using pickle.load(). The model predicts the likelihood of heart disease based on the input features provided by the user.

Displaying Prediction Results:

The prediction result (presence or absence of heart disease) is displayed to the user through the Streamlit web interface, providing valuable insights into the individual's health condition based on the predictive model.

Conclusion

Summary of Findings:

Output: The model performed well with an accuracy of 85%, primarily driven by age, cholesterol, and heart rate.

Explanation: Insights from the model underscored the importance of physiological indicators in predicting heart disease, validating the model's efficacy.

Explanation video: <https://nileuniversity-my.sharepoint.com/:v:/g/personal/j_tamer2163_nu_edu_eg/EQr6kvRuapRAkc4Oyq1gsoEBMuwuEcdr5nn34bBqpVJuEA?referrer=Teams.TEAMS-ELECTRON&referrerScenario=MeetingChicletGetLink.view.view>