Heart Disease Prediction Project

Project Overview

This project aims to predict the presence of heart disease in patients using machine learning techniques. The model is trained on a dataset containing various medical attributes and is capable of classifying whether a patient is likely to have heart disease.

Dataset

- **File:** `dataset.csv`

Attribute Description

Attribute	Code/Column Name	Unit	Data Type	Description/Values
Age	age	years	Numeric	Patient age in years
Sex	sex	-	Binary	0 = female, 1 = male
Chest pain type	chest pain type	-	Nominal	1 = typical angina, 2 = atypypical angina, 3 = non-anginal pain, 4 = asymptomatic
Resting blood pressure	resting bp s	mm Hg	Numeric	Resting blood pressure
Serum cholesterol	cholesterol	mg/dl	Numeric	Serum cholesterol
Fasting blood sugar	fasting blood sugar	-	Binary	1 = sugar > 120mg/dL, 0 = sugar < 120mg/dL
Resting electrocardiogram	resting ecg	-	Nominal	0 = normal, 1 = ST-T wave abnormality, 2 = LV hypertrophy by Estes' criteria
Maximum heart rate	max heart rate	bpm	Numeric	Maximum heart rate achieved
Exercise induced angina	exercise angina	-	Binary	0 = no, 1 = yes
ST depression (oldpeak)	oldpeak	-	Numeric	ST depression induced by exercise relative to rest
Slope of ST segment	ST slope	-	Nominal	1 = upward, 2 = flat, 3 = downward
Target	target	-	Binary	0 = Normal, 1 = Heart Disease

- **Features:**
  - age
  - sex
  - chest pain type
  - resting bp s
  - cholesterol
  - fasting blood sugar
  - resting ecg
  - max heart rate
  - exercise angina
  - oldpeak
  - ST slope
- **Target:**
  - `target` (1 = heart disease, 0 = no heart disease)

Methodology

- **Preprocessing:**
  - Loaded the dataset and selected features.
  - One-hot encoded categorical features.
  - Split the data into training and test sets (80/20 split).
- **Model:**
  - Used a Random Forest Classifier.
  - Performed hyperparameter tuning using GridSearchCV with 5-fold cross-validation.
  - Parameters tuned: `n_estimators`, `max_depth`, `min_samples_split`.
- **Evaluation:**
  - Evaluated the best model on the test set using accuracy and classification report.

Results

- **Best Parameters:** `{ 'max_depth': None, 'min_samples_split': 2, 'n_estimators': 200 }`
- **Test Accuracy:** `0.937`
- **Classification Report:**

```
              precision    recall  f1-score   support

           0       0.93      0.94      0.93       112
           1       0.94      0.94      0.94       126

    accuracy                           0.94       238
   macro avg       0.94      0.94      0.94       238
weighted avg       0.94      0.94      0.94       238
```

- **Model Saved:** `heart_disease_model.joblib`

Usage

1. Place the dataset (`dataset.csv`) in the project directory.
2. Run the script:

   ```
   python heart_disease_prediction.py
   ```

3. The script will output the best parameters, test accuracy, and save the trained model as `heart_disease_model.joblib`.

Requirements

- Python 3.x
- pandas
- scikit-learn
- joblib

Install dependencies with:

```
pip install pandas scikit-learn joblib
```

Notes

- The script can be modified to include additional preprocessing or feature engineering as needed.
- For more details, see the script `heart_disease_prediction.py`.