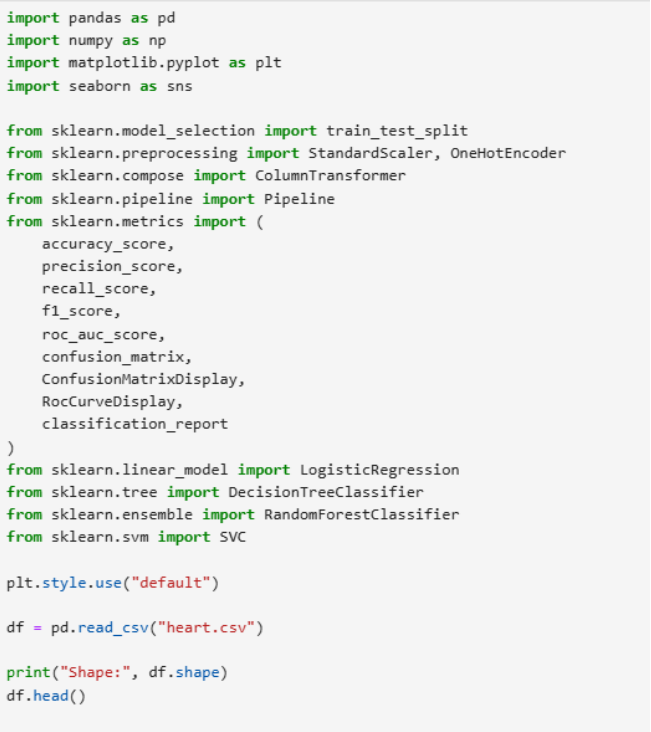
Heart Disease Prediction Using Machine Learning

Ruhil Patel

Demo Video Link | Github Link

1. **Project Statement**
   The aim of this project is to design and test various machine learning classification models that can predict the presence of heart disease, as represented by the target variable, using several clinical and demographic features. This dataset consists of 1,025 instances with 14 features, such as age, sex, chest pain type, cholesterol, and maximum heart rate achieved. The main task is to develop a model that will provide high predictive performance, especially in terms of recall, which is critical in a clinical setting to avoid missed cases of heart disease.
2. **Novelty and Importance**
   The purpose of this project is to develop an accurate and reliable way to predict heart disease early, which is important because if a person has heart disease diagnosed early, doctors can act sooner and generally see better results.
   Importance: Cardiovascular Disease still ranks among the leading causes of death worldwide, so creating reliable, readily available, high-recall models to help predict these conditions can help clinics make decisions using non-invasive methods.
   Novelty: Although predicting heart disease is one of the more prevalent tasks for Machine Learning, this report's major contribution lies in its comparative approach to evaluating multiple model types (Logistic Regression, Decision Tree, Random Forest, Support Vector Machine), and in the way it strategically used both cross-validation and hyperparameter tuning to maintain the integrity of model performance. In addition, providing an analysis of feature importance enables clinicians to identify the most significant risk factors and potentially makes the models more interpretable and actionable.
3. **Progress and Contribution**
   As the project advanced through the phases of typical machine learning workflow, a finished, adjusted, and persistent model was produced.
   - **Data Loading and Exploration (EDA):** I have loaded all available information, checked for any missing values (none); examined the type of features in the data set; created descriptive statistics on the original data set; and created an age group column for categorical purposes.



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.metrics import (
    accuracy_score,
    precision_score,
    recall_score,
    f1_score,
    roc_auc_score,
    confusion_matrix,
    ConfusionMatrixDisplay,
    RocCurveDisplay,
    classification_report
)
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC

plt.style.use("default")

df = pd.read_csv("heart.csv")

print("Shape:", df.shape)
df.head()
```

Figure 1: Data Loading

- **Visualization:** I conducted an analysis of how the age distributions affect heart disease rates based on age groupings, how cholesterol affects heart disease based on the age groupings, and how cholesterol relates to the target variable. I created a heat map to visualize the correlation between the numeric features in the data set.
- **Data Preprocessing:** I also split the training and test data using stratification to preserve the class distribution during the splitting process; set up a ColumnTransformer that will apply the StandardScaler method to numeric features, and the OneHotEncoder method to the categorical features in the data set.
- **Model Training & Evaluation (Baseline):** Next, I trained four initial baseline classification models (Logistic Regression, Decision Tree, Random Forest, and SVM) using a Pipeline for end-to-end processing, including both the development of the classification models as well as the application of preprocessing methods. I evaluated each of the models on various metrics (Accuracy, Precision, Recall, F1 Score, and ROC-AUC) using the test data.
- **Model Selection & Analysis:** After determining Random Forest to be the first best classification model, based on the highest ROC-AUC, I then applied both cross-validation to evaluate Random Forest and using Permutation and Native methods for Feature Importance Analysis.
- **Hyperparameter Tuning:** Finally, I used the GridSearchCV method to optimize the tuned Random Forest and Logistic Regression models using a 5-fold Cross Validation method and a roc_auc scoring method.

- **Finalization:** I saved the tuned Random Forest model as a serialized (.pkl) file containing the following data. (heart_disease_model.pkl).

Individual Contribution
I'm responsible for conducting and completing aspects of project including, but not limited to, analyzing data, creating a feature engineering (feature age adults), building pipelines, running comparative model training, performing detailed evaluation of performance and hyperparameter tuning, determining feature importance, and writing final report.

4. Models and Algorithms

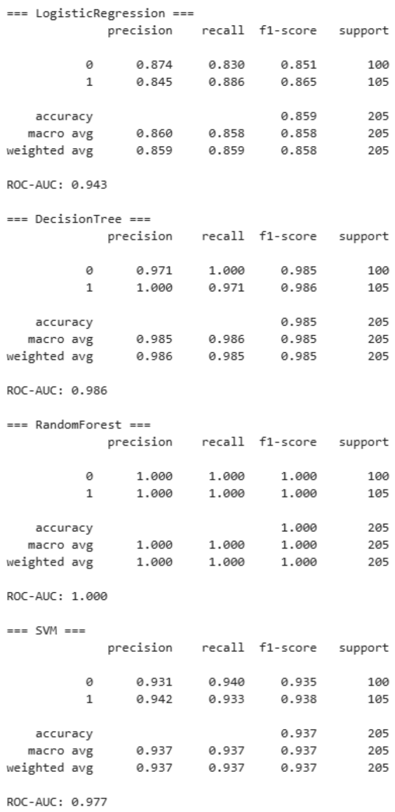
Four main classification methods that were all incorporated into a sklearn.pipeline were assessed.pipeline that includes the preparatory stages.

Preprocessing Pipeline
Data preparation is ensured by the ColumnTransformer, often known as preprocess:

- **Numeric Columns:** ['age', 'resting_blood_pressure', 'cholestoral', 'Max_heart_rate', 'oldpeak'] are scaled using StandardScaler.
- **Categorical Columns:** ['sex', 'chest_pain_type', 'fasting_blood_sugar', 'rest_ecg', 'exercise_induced_angina', 'slope', 'vessels_colored_by_flourosopy', 'thalassemia', 'age_group'] are encoded using OneHotEncoder

Classification Models

1. **Logistic Regression (LogisticRegression):** This model functions as a baseline and is linear and interpretable. L1/L2 regularization (penalty) and regularization strength (C) are used to optimize it.
2. **Decision Tree (DecisionTreeClassifier):** This is a non-linear and interpretable tree-structured classifier.
3. **Random Forest (RandomForestClassifier):** A Random Forest Classifier employs an ensemble of many different decision trees (Bagging). A Random Forest Classifier is considered one of the most accurate and stable types of models. It is optimized with n_estimators, max_depth, min_samples_split, and min_samples_leaf.
4. **Support Vector Machine (SVC):** This is a classifier that is based on a discriminative model, using a non-linear kernel (rbf) to identify the best hyperplane.



```
=== LogisticRegression ===
              precision    recall  f1-score   support

           0      0.874     0.830     0.851       100
           1      0.845     0.886     0.865       105

    accuracy                          0.859       205
   macro avg      0.860     0.858     0.858       205
weighted avg      0.859     0.859     0.858       205

ROC-AUC: 0.943

=== DecisionTree ===
              precision    recall  f1-score   support

           0      0.971     1.000     0.985       100
           1      1.000     0.971     0.986       105

    accuracy                          0.985       205
   macro avg      0.985     0.986     0.985       205
weighted avg      0.986     0.985     0.985       205

ROC-AUC: 0.986

=== RandomForest ===
              precision    recall  f1-score   support

           0      1.000     1.000     1.000       100
           1      1.000     1.000     1.000       105

    accuracy                          1.000       205
   macro avg      1.000     1.000     1.000       205
weighted avg      1.000     1.000     1.000       205

ROC-AUC: 1.000

=== SVM ===
              precision    recall  f1-score   support

           0      0.931     0.940     0.935       100
           1      0.942     0.933     0.938       105

    accuracy                          0.937       205
   macro avg      0.937     0.937     0.937       205
weighted avg      0.937     0.937     0.937       205

ROC-AUC: 0.977
```

Figure 2: Models

5. Experimental Design

Data Split: 80% of the dataset was used for training, while 20% was used for testing. To preserve an equal percentage of heart disease cases (target=1) in the training and testing datasets, we carried out the stratification using stratify = y. This lessens the possibility of prejudice resulting from class disparity.

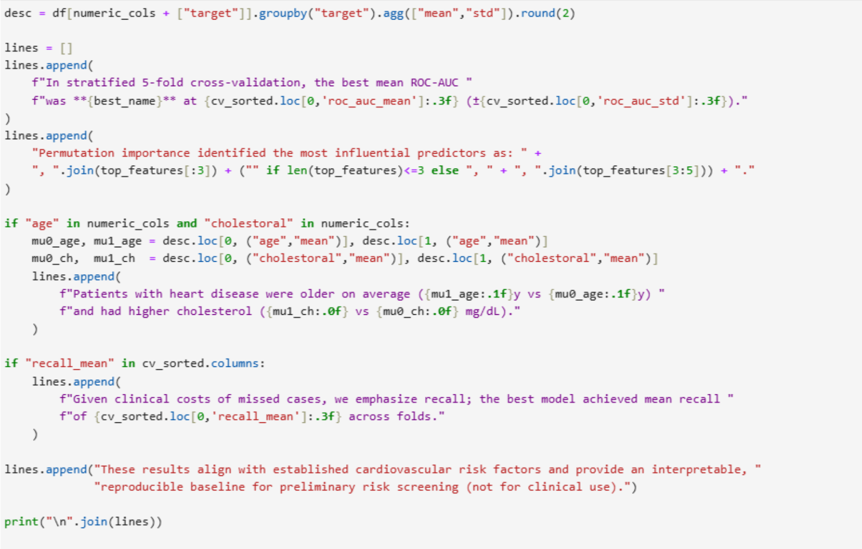
Metrics for Evaluation: The evaluation metrics, which were primarily concerned with the models' overall performance and clinical value, comprised: The main assessment tool we will employ to compare and assess the performance of our models is ROC-AUC (Area Under the Receiver Operating Characteristic Curve). It shows the model's ability to differentiate between class labels.

Recall (Sensitivity): It is impossible to exaggerate the significance of having a high recall when making medical decisions. Reducing false negatives, the possibility of missing a diagnosis of heart disease is the aim in medicine.

The most often computed metrics that give a general idea of how well the categorization is doing overall are accuracy, precision, and F1 score.

5-Fold Stratified Cross-Validation (CV) Robustness Testing: I used this technique to evaluate the consistency of our models across several subsets of the training dataset. We present the mean and standard deviation for every measure for every model across all five folds (Execution Count 12).

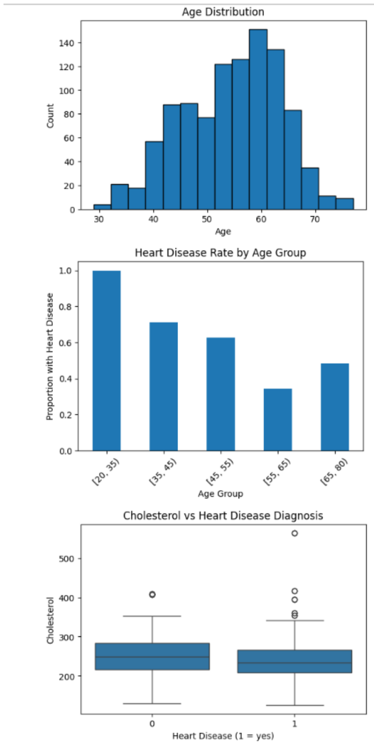
Hyperparameters: To avoid overfitting and maximizing generalization performance, we employed GridSearchCV with 5-Fold CV on our training dataset to determine the ideal hyperparameters for each of the top-performing models (Logistic Regression and Random Forest).



```python
desc = df[numeric_cols + ["target"]].groupby("target").agg(["mean","std"]).round(2)

lines = []
lines.append(
    f"In stratified 5-fold cross-validation, the best mean ROC-AUC "
    f"was **{best_name}** at {cv_sorted.loc[0,'roc_auc_mean']:.3f} (±{cv_sorted.loc[0,'roc_auc_std']:.3f})."
)
lines.append(
    "Permutation importance identified the most influential predictors as: " +
    ", ".join(top_features[:3]) + ("" if len(top_features)<=3 else ", " + ", ".join(top_features[3:5])) + "."
)

if "age" in numeric_cols and "cholestoral" in numeric_cols:
    mu0_age, mu1_age = desc.loc[0, ("age","mean")], desc.loc[1, ("age","mean")]
    mu0_ch,  mu1_ch  = desc.loc[0, ("cholestoral","mean")], desc.loc[1, ("cholestoral","mean")]
    lines.append(
        f"Patients with heart disease were older on average ({mu1_age:.1f}y vs {mu0_age:.1f}y) "
        f"and had higher cholesterol ({mu1_ch:.0f} vs {mu0_ch:.0f} mg/dL)."
    )

if "recall_mean" in cv_sorted.columns:
    lines.append(
        f"Given clinical costs of missed cases, we emphasize recall; the best model achieved mean recall "
        f"of {cv_sorted.loc[0,'recall_mean']:.3f} across folds."
    )

lines.append("These results align with established cardiovascular risk factors and provide an interpretable, "
             "reproducible baseline for preliminary risk screening (not for clinical use).")

print("\n".join(lines))
```

Figure 3: Experimental Design

6. Screenshots of Outputs



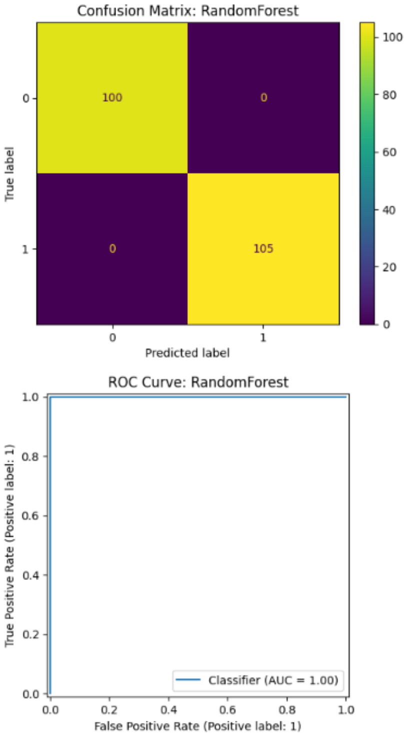


Figure 4: Age Group

Figure 5: Random Forest

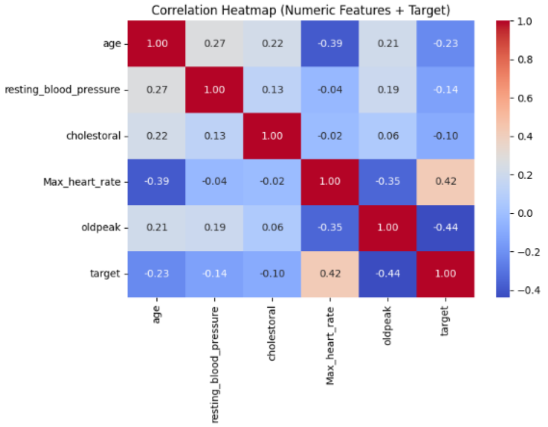


Figure 6: Correlation Heatmap

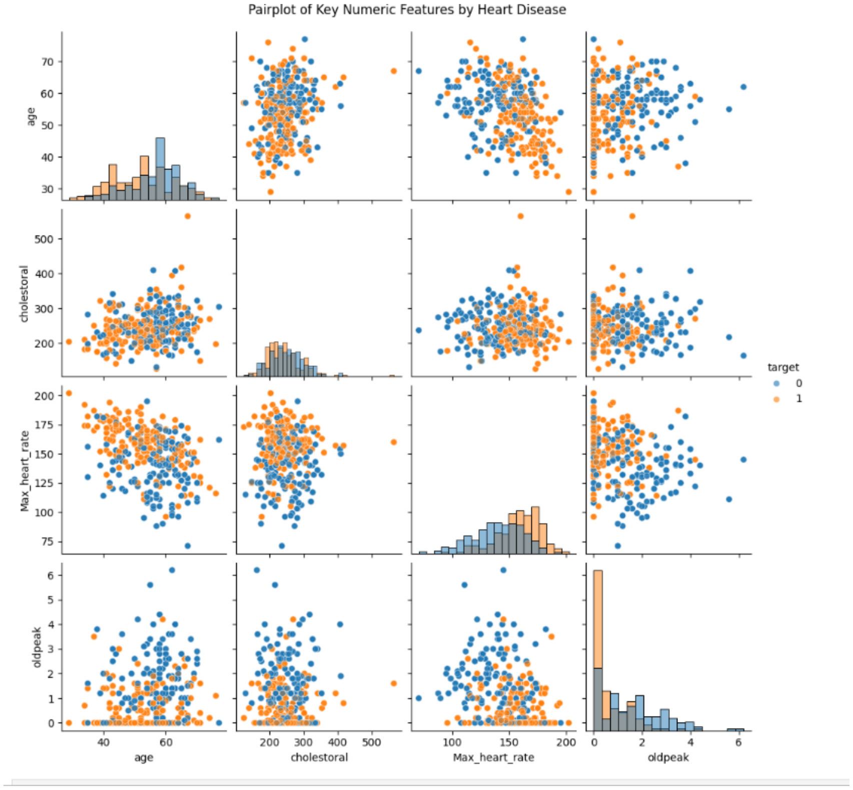
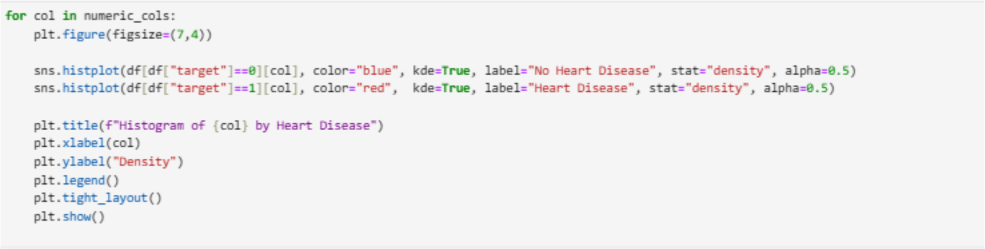


Figure 7: Key Numeric Features



```
for col in numeric_cols:
    plt.figure(figsize=(7,4))

    sns.histplot(df[df["target"]==0][col], color="blue", kde=True, label="No Heart Disease", stat="density", alpha=0.5)
    sns.histplot(df[df["target"]==1][col], color="red",  kde=True, label="Heart Disease", stat="density", alpha=0.5)

    plt.title(f"Histogram of {col} by Heart Disease")
    plt.xlabel(col)
    plt.ylabel("Density")
    plt.legend()
    plt.tight_layout()
    plt.show()
```

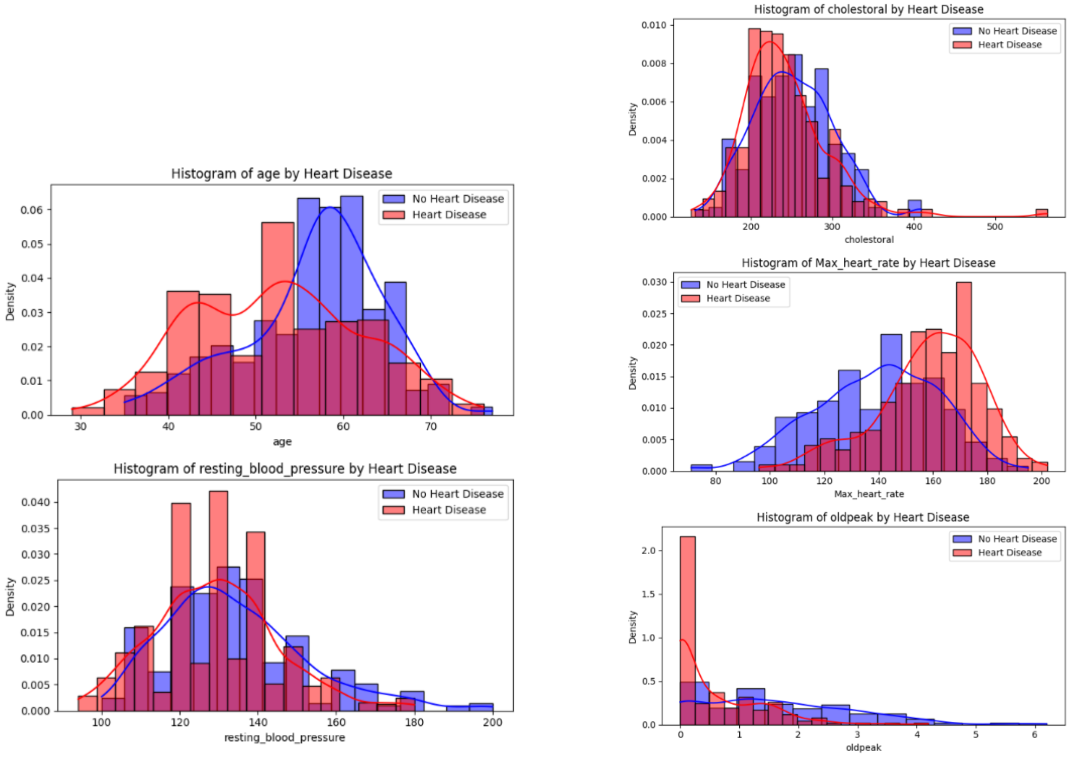
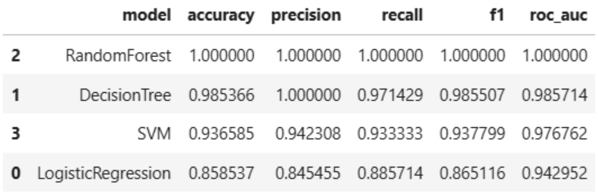


Figure 8: Histogram

7. Detailed Analysis of Results and Key Results and Evaluation

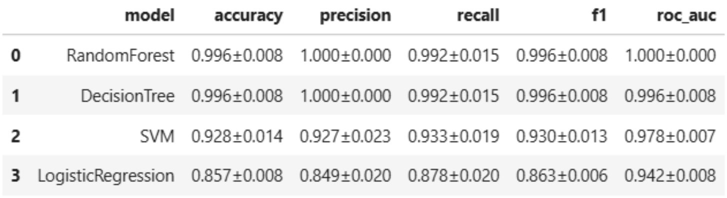
Baseline Model Comparison (Test Set):



	model	accuracy	precision	recall	f1	roc_auc
2	RandomForest	1.000000	1.000000	1.000000	1.000000	1.000000
1	DecisionTree	0.985366	1.000000	0.971429	0.985507	0.985714
3	SVM	0.936585	0.942308	0.933333	0.937799	0.976762
0	LogisticRegression	0.858537	0.845455	0.885714	0.865116	0.942952

The Random Forest Model attained perfect scores (1.000-Accuracy, F1, and ROC-AUC) on the test data set and thus indicates that the Random Forest Model is likely to be the best model based on accuracy; however, it does raise concerns for overfitting or data leakage because of the small and specific composition of the combined dataset.

Cross-Validation Results (Robustness)



	model	accuracy	precision	recall	f1	roc_auc
0	RandomForest	0.996±0.008	1.000±0.000	0.992±0.015	0.996±0.008	1.000±0.000
1	DecisionTree	0.996±0.008	1.000±0.000	0.992±0.015	0.996±0.008	0.996±0.008
2	SVM	0.928±0.014	0.927±0.023	0.933±0.019	0.930±0.013	0.978±0.007
3	LogisticRegression	0.857±0.008	0.849±0.020	0.878±0.020	0.863±0.006	0.942±0.008

According to the 5-fold Cross Validation results, Random Forest is identified as the most resilient and powerful model (with a mean ROC AUC = 1.000 ± 0.000 and a total mean recall of = 0.992 (±0.015)) across alternative data partitions in support of the primary focus of clinical applications, which prioritises high recall; therefore, Random Forest would be the ideal candidate model for this project.

8. Conclusion

Through this project, I built several different ways to predict heart disease and evaluated the models for performance. We discovered that the Random Forest classifier trained on this dataset produced the best results overall with a mean ROC-AUC score of 1.000 (±0.000) and mean recall of 0.992 (±0.015), using 5-fold stratified cross-validation. These recall metrics are particularly important because they allow people conducting initial clinical screenings to identify as many cases as possible.

The Random Forest classifier was able to identify risk factors associated with heart disease and showed that there are many key features that affect whether someone has heart disease, when ranked by how much they contribute: Max_heart_rate; oldpeak; age. I have saved a version of the optimized Random Forest classifier for future use in an interpretive reproducible manner for developing clinical support tools.