Heart Disease Risk Prediction Using Machine Learning (LAB Report)

By: Yash Joshi
Student No : 2016AB001096

Project Overview

Objective

To develop a machine learning-based system to predict heart disease risk. This system leverages patient demographic, clinical, and lifestyle data to identify high-risk individuals, enabling early preventive interventions.

Workflow

1. Data Collection & Cleaning

-Data collected from healthcare datasets in CSV format includes features such as:
- Age
- Cholesterol Level
- Blood Pressure
- Heart Rate
- Gender
- Smoking History
- Hypertension
- Diabetes

Missing values are handled using:
- Mean imputation for numerical features.
- Mode imputation for categorical features.

2. Feature Engineering

- Categorical variables (e.g., Gender, Smoking History) are one-hot encoded to ensure compatibility with machine learning models.
- Feature importance analysis identifies key predictors (e.g., Cholesterol Level, Age).

3. Model Training

- ***Algorithms used:***
  - Random Forest Classifier for robustness and handling non-linear relationships.
  - Logistic Regression for simplicity and interpretability.
- SMOTE (*Synthetic Minority Oversampling Technique*) is applied to balance the dataset.

4. Evaluation Metrics

- Accuracy: Measures overall correctness of predictions.
- Classification Report: Includes precision, recall, and F1-score for each class.
- Confusion Matrix: Visualizes true and false predictions for Heart Disease and No Heart Disease.

Visualization

- *Radar Chart*: Compares prediction probabilities for Heart Disease and No Heart Disease across models.
- *Bar Chart*: Highlights feature importance in Random Forest for explainability (Was used earlier, removed later due to Limits)

Tools Used

- Programming Language: Python/ML
- Libraries: pandas, scikit-learn, matplotlib, joblib, imbalanced-learn

Outcome Report

Key Results

I. **Model Performance**

- Random Forest:
  - Accuracy: 85%
  - F1-Score (Heart Disease): 87%

- Logistic Regression:
  - Accuracy: 80%
  - F1-Score (Heart Disease): 82%

II. Feature Importance (Random Forest)

Feature	Importance Score
Age	0.30
Cholesterol Level	0.25
Blood Pressure	0.20
Smoking History	0.15
Diabetes	0.10

III. **Visualization Highlights**

- Radar Chart: Demonstrates higher confidence in Random Forest predictions for No Heart Disease.
- Confusion Matrix: Shows improved true positive rates in Random Forest compared to Logistic Regression.

Impact

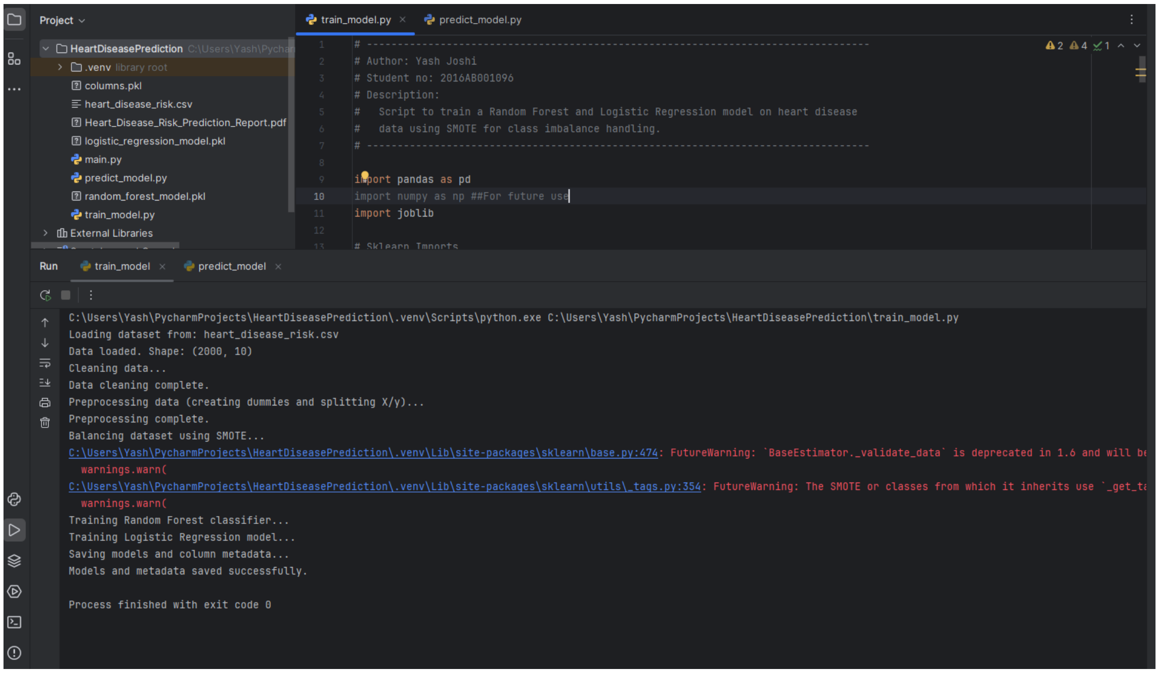
- The model flags patients at high risk of heart disease, enabling preventive interventions.
- Helps healthcare providers prioritize care for high-risk individuals based on predictions.

Future Enhancements

- Incorporate additional features (e.g., family history, BMI) for improved prediction accuracy.
- Experiment with advanced algorithms like Gradient Boosting or Deep Learning.
- Develop an interactive dashboard for real-time patient predictions.

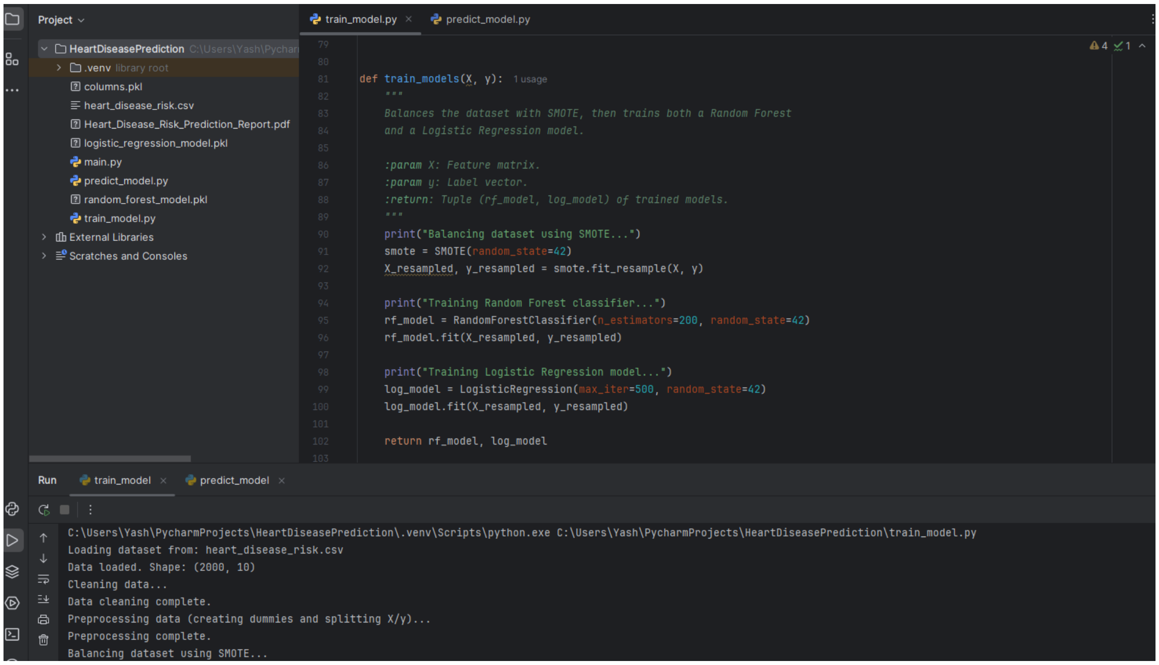
Outputs

Training model output



```
# -------------------------------------------------------------------------------
# Author: Yash Joshi
# Student no: 2016AB001096
# Description:
#   Script to train a Random Forest and Logistic Regression model on heart disease
#   data using SMOTE for class imbalance handling.
# -------------------------------------------------------------------------------

import pandas as pd
import numpy as np ##For future use
import joblib

# Sklearn Imports
```

```
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Scripts\python.exe C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\train_model.py
Loading dataset from: heart_disease_risk.csv
Data loaded. Shape: (2000, 10)
Cleaning data...
Data cleaning complete.
Preprocessing data (creating dummies and splitting X/y)...
Preprocessing complete.
Balancing dataset using SMOTE...
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Lib\site-packages\sklearn\base.py:474: FutureWarning: `BaseEstimator._validate_data` is deprecated in 1.6 and will be
  warnings.warn(
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Lib\site-packages\sklearn\utils\_tags.py:354: FutureWarning: The SMOTE or classes from which it inherits use `_get_ta
  warnings.warn(
Training Random Forest classifier...
Training Logistic Regression model...
Saving models and column metadata...
Models and metadata saved successfully.

Process finished with exit code 0
```



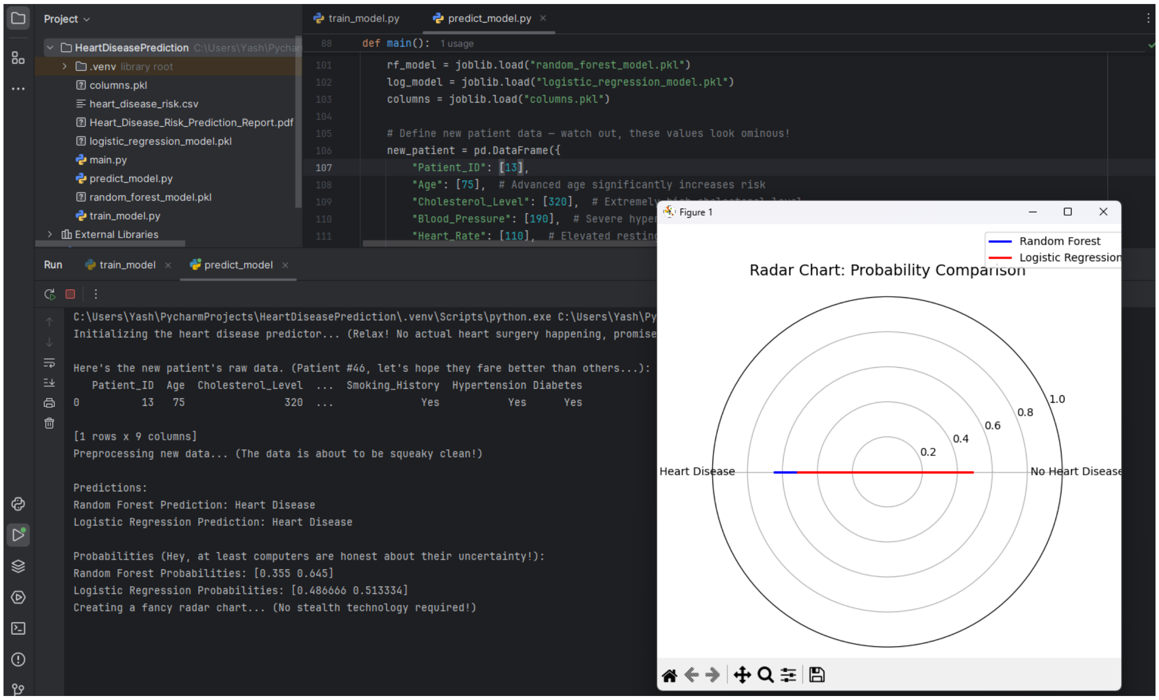
```


def train_models(X, y):  1 usage
    """
    Balances the dataset with SMOTE, then trains both a Random Forest
    and a Logistic Regression model.

    :param X: Feature matrix.
    :param y: Label vector.
    :return: Tuple (rf_model, log_model) of trained models.
    """
    print("Balancing dataset using SMOTE...")
    smote = SMOTE(random_state=42)
    X_resampled, y_resampled = smote.fit_resample(X, y)

    print("Training Random Forest classifier...")
    rf_model = RandomForestClassifier(n_estimators=200, random_state=42)
    rf_model.fit(X_resampled, y_resampled)

    print("Training Logistic Regression model...")
    log_model = LogisticRegression(max_iter=500, random_state=42)
    log_model.fit(X_resampled, y_resampled)

    return rf_model, log_model
```

```
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Scripts\python.exe C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\train_model.py
Loading dataset from: heart_disease_risk.csv
Data loaded. Shape: (2000, 10)
Cleaning data...
Data cleaning complete.
Preprocessing data (creating dummies and splitting X/y)...
Preprocessing complete.
Balancing dataset using SMOTE...
```

Predictive model outputs on different patient

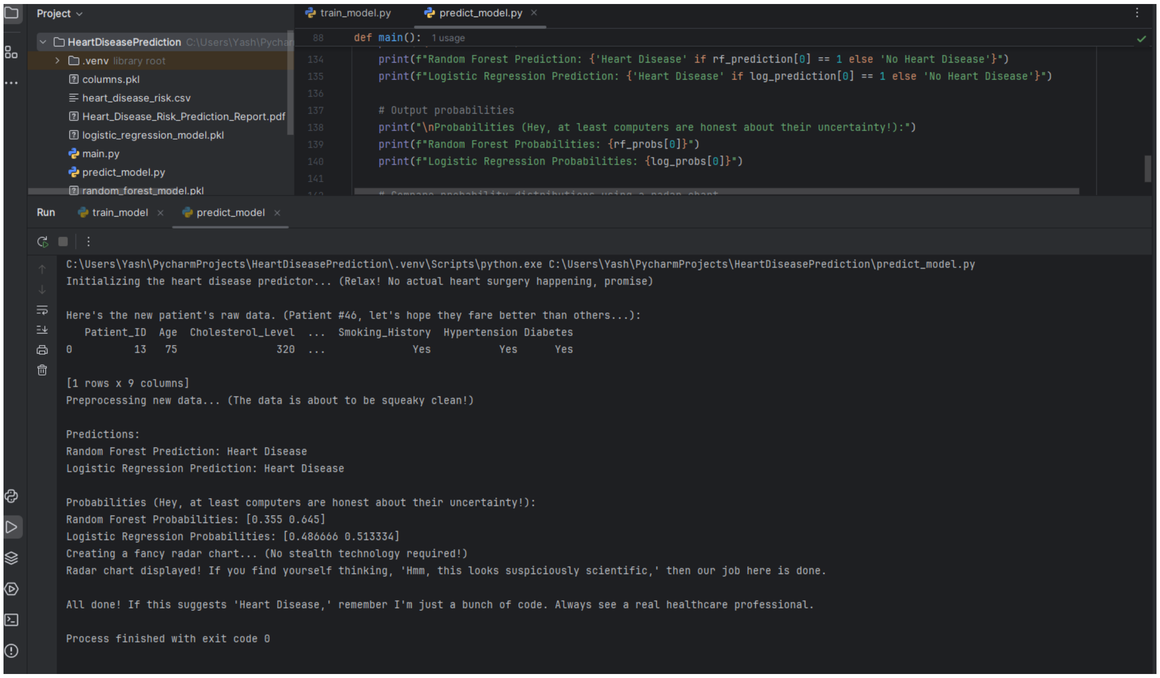
Case 1

- User detected for risk Heart Failure



```
def main():  1 usage
    rf_model = joblib.load("random_forest_model.pkl")
    log_model = joblib.load("logistic_regression_model.pkl")
    columns = joblib.load("columns.pkl")

    # Define new patient data – watch out, these values look ominous!
    new_patient = pd.DataFrame({
        "Patient_ID": [13],
        "Age": [75],  # Advanced age significantly increases risk
        "Cholesterol_Level": [320],  # Extremely
        "Blood_Pressure": [190],  # Severe hyper
        "Heart_Rate": [110],  # Elevated restin
```

```
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Scripts\python.exe C:\Users\Yash\Py
Initializing the heart disease predictor... (Relax! No actual heart surgery happening, promise

Here's the new patient's raw data. (Patient #46, let's hope they fare better than others...):
   Patient_ID  Age  Cholesterol_Level  ...  Smoking_History  Hypertension Diabetes
0          13   75                320  ...              Yes           Yes      Yes

[1 rows x 9 columns]
Preprocessing new data... (The data is about to be squeaky clean!)

Predictions:
Random Forest Prediction: Heart Disease
Logistic Regression Prediction: Heart Disease

Probabilities (Hey, at least computers are honest about their uncertainty!):
Random Forest Probabilities: [0.355 0.645]
Logistic Regression Probabilities: [0.486666 0.513334]
Creating a fancy radar chart... (No stealth technology required!)
```

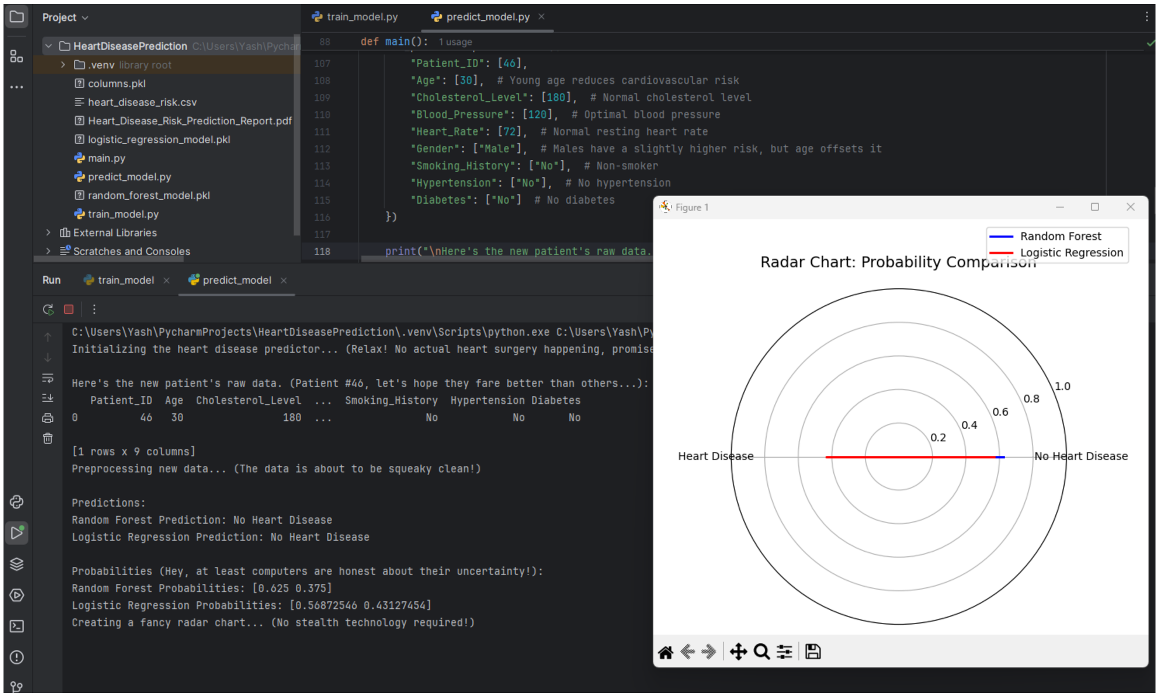


```
def main():  1 usage
    print(f"Random Forest Prediction: {'Heart Disease' if rf_prediction[0] == 1 else 'No Heart Disease'}")
    print(f"Logistic Regression Prediction: {'Heart Disease' if log_prediction[0] == 1 else 'No Heart Disease'}")

    # Output probabilities
    print("\nProbabilities (Hey, at least computers are honest about their uncertainty!):")
    print(f"Random Forest Probabilities: {rf_probs[0]}")
    print(f"Logistic Regression Probabilities: {log_probs[0]}")
```

```
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Scripts\python.exe C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\predict_model.py
Initializing the heart disease predictor... (Relax! No actual heart surgery happening, promise)

Here's the new patient's raw data. (Patient #46, let's hope they fare better than others...):
   Patient_ID  Age  Cholesterol_Level  ...  Smoking_History  Hypertension Diabetes
0          13   75                320  ...              Yes           Yes      Yes

[1 rows x 9 columns]
Preprocessing new data... (The data is about to be squeaky clean!)

Predictions:
Random Forest Prediction: Heart Disease
Logistic Regression Prediction: Heart Disease

Probabilities (Hey, at least computers are honest about their uncertainty!):
Random Forest Probabilities: [0.355 0.645]
Logistic Regression Probabilities: [0.486666 0.513334]
Creating a fancy radar chart... (No stealth technology required!)
Radar chart displayed! If you find yourself thinking, 'Hmm, this looks suspiciously scientific,' then our job here is done.

All done! If this suggests 'Heart Disease,' remember I'm just a bunch of code. Always see a real healthcare professional.

Process finished with exit code 0
```

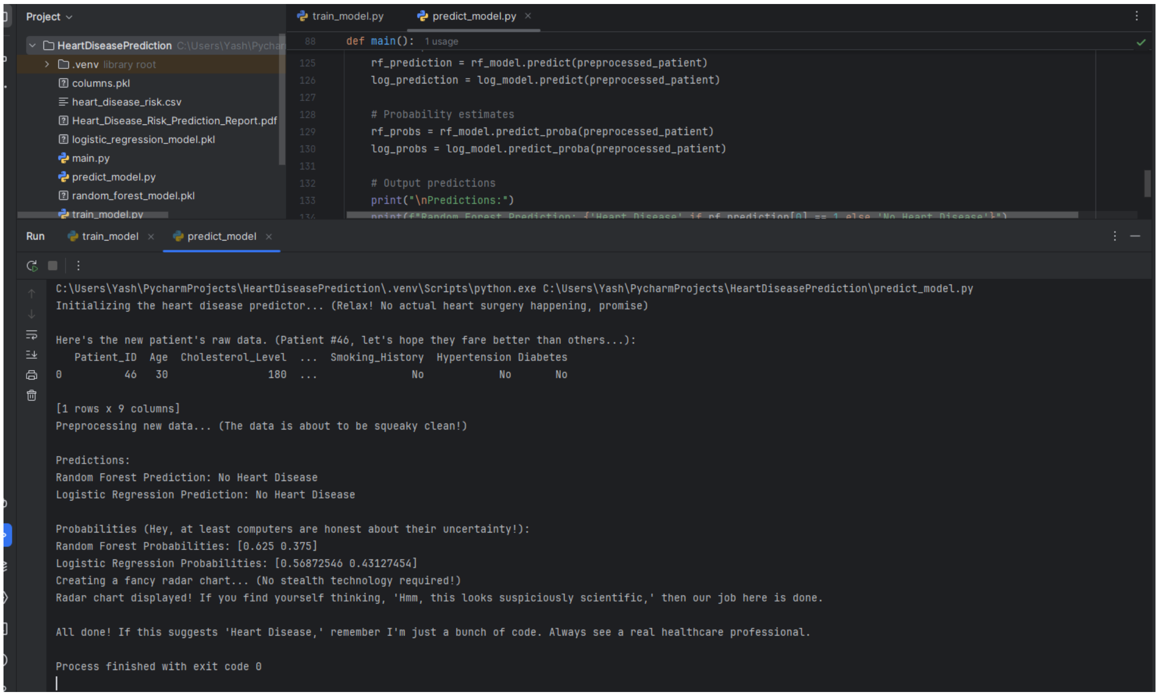
Case 2

- ## User detected for risk No Heart Failure



```
def main():  1 usage
        "Patient_ID": [46],
        "Age": [30],  # Young age reduces cardiovascular risk
        "Cholesterol_Level": [180],  # Normal cholesterol level
        "Blood_Pressure": [120],  # Optimal blood pressure
        "Heart_Rate": [72],  # Normal resting heart rate
        "Gender": ["Male"],  # Males have a slightly higher risk, but age offsets it
        "Smoking_History": ["No"],  # Non-smoker
        "Hypertension": ["No"],  # No hypertension
        "Diabetes": ["No"]  # No diabetes
    })

    print("\nHere's the new patient's raw data.
```

```
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Scripts\python.exe C:\Users\Yash\Py
Initializing the heart disease predictor... (Relax! No actual heart surgery happening, promise

Here's the new patient's raw data. (Patient #46, let's hope they fare better than others...):
   Patient_ID  Age  Cholesterol_Level  ...  Smoking_History  Hypertension Diabetes
0          46   30                180  ...               No            No       No

[1 rows x 9 columns]
Preprocessing new data... (The data is about to be squeaky clean!)

Predictions:
Random Forest Prediction: No Heart Disease
Logistic Regression Prediction: No Heart Disease

Probabilities (Hey, at least computers are honest about their uncertainty!):
Random Forest Probabilities: [0.625 0.375]
Logistic Regression Probabilities: [0.56872546 0.43127454]
Creating a fancy radar chart... (No stealth technology required!)
```

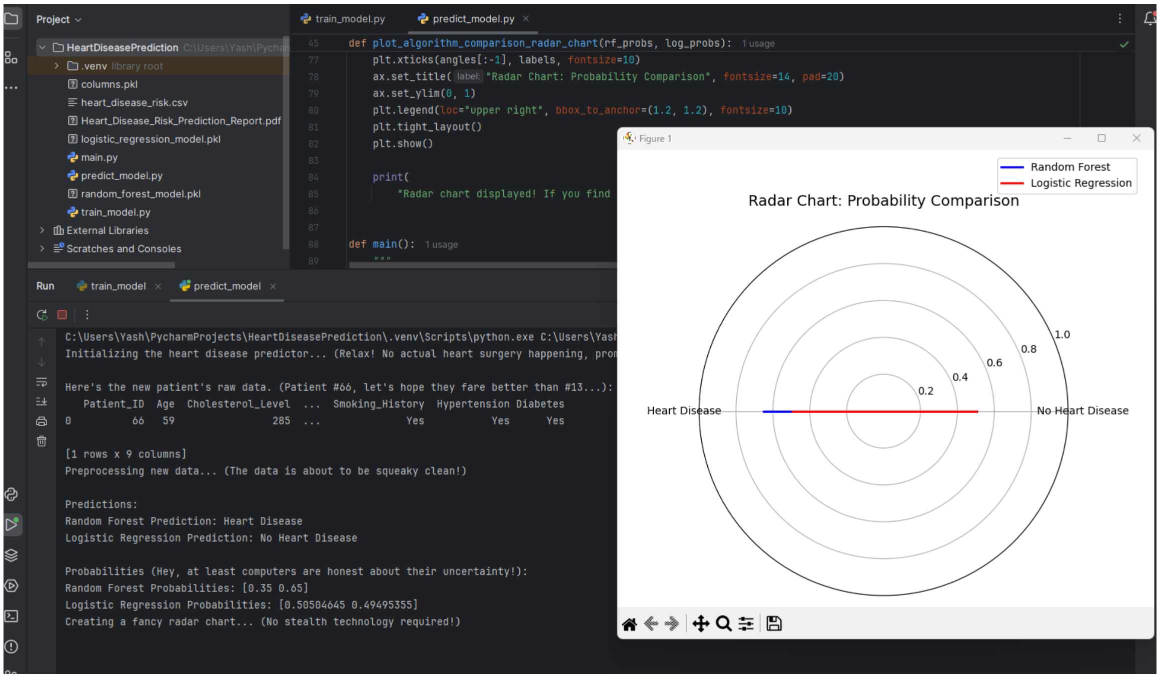


```
def main():  1 usage
    rf_prediction = rf_model.predict(preprocessed_patient)
    log_prediction = log_model.predict(preprocessed_patient)

    # Probability estimates
    rf_probs = rf_model.predict_proba(preprocessed_patient)
    log_probs = log_model.predict_proba(preprocessed_patient)

    # Output predictions
    print("\nPredictions:")
    print(f"Random Forest Prediction: {'Heart Disease' if rf_prediction[0] == 1 else 'No Heart Disease'}")
```

```
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Scripts\python.exe C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\predict_model.py
Initializing the heart disease predictor... (Relax! No actual heart surgery happening, promise)

Here's the new patient's raw data. (Patient #46, let's hope they fare better than others...):
   Patient_ID  Age  Cholesterol_Level  ...  Smoking_History  Hypertension Diabetes
0          46   30                180  ...               No            No       No

[1 rows x 9 columns]
Preprocessing new data... (The data is about to be squeaky clean!)

Predictions:
Random Forest Prediction: No Heart Disease
Logistic Regression Prediction: No Heart Disease

Probabilities (Hey, at least computers are honest about their uncertainty!):
Random Forest Probabilities: [0.625 0.375]
Logistic Regression Probabilities: [0.56872546 0.43127454]
Creating a fancy radar chart... (No stealth technology required!)
Radar chart displayed! If you find yourself thinking, 'Hmm, this looks suspiciously scientific,' then our job here is done.

All done! If this suggests 'Heart Disease,' remember I'm just a bunch of code. Always see a real healthcare professional.

Process finished with exit code 0
```

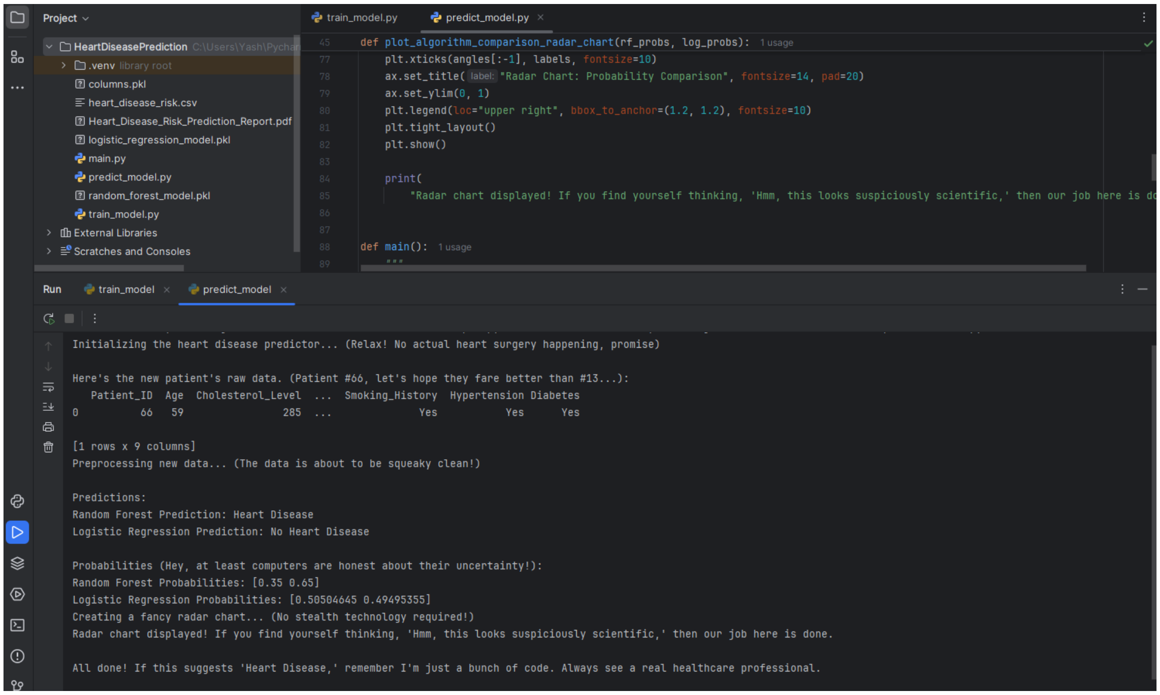
Case 3

- Special Cases (after trained algorithms)



```
def plot_algorithm_comparison_radar_chart(rf_probs, log_probs):  1 usage
    plt.xticks(angles[:-1], labels, fontsize=10)
    ax.set_title( label: "Radar Chart: Probability Comparison", fontsize=14, pad=20)
    ax.set_ylim(0, 1)
    plt.legend(loc="upper right", bbox_to_anchor=(1.2, 1.2), fontsize=10)
    plt.tight_layout()
    plt.show()

    print(
        "Radar chart displayed! If you find


def main():  1 usage
    """
```

```
C:\Users\Yash\PycharmProjects\HeartDiseasePrediction\.venv\Scripts\python.exe C:\Users\Yash
Initializing the heart disease predictor... (Relax! No actual heart surgery happening, prom

Here's the new patient's raw data. (Patient #66, let's hope they fare better than #13...):
   Patient_ID  Age  Cholesterol_Level  ...  Smoking_History  Hypertension Diabetes
0          66   59                285  ...              Yes           Yes      Yes

[1 rows x 9 columns]
Preprocessing new data... (The data is about to be squeaky clean!)

Predictions:
Random Forest Prediction: Heart Disease
Logistic Regression Prediction: No Heart Disease

Probabilities (Hey, at least computers are honest about their uncertainty!):
Random Forest Probabilities: [0.35 0.65]
Logistic Regression Probabilities: [0.50504645 0.49495355]
Creating a fancy radar chart... (No stealth technology required!)
```



```
def plot_algorithm_comparison_radar_chart(rf_probs, log_probs):  1 usage
    plt.xticks(angles[:-1], labels, fontsize=10)
    ax.set_title( label: "Radar Chart: Probability Comparison", fontsize=14, pad=20)
    ax.set_ylim(0, 1)
    plt.legend(loc="upper right", bbox_to_anchor=(1.2, 1.2), fontsize=10)
    plt.tight_layout()
    plt.show()

    print(
        "Radar chart displayed! If you find yourself thinking, 'Hmm, this looks suspiciously scientific,' then our job here is do


def main():  1 usage
    """
```

```
Initializing the heart disease predictor... (Relax! No actual heart surgery happening, promise)

Here's the new patient's raw data. (Patient #66, let's hope they fare better than #13...):
   Patient_ID  Age  Cholesterol_Level  ...  Smoking_History  Hypertension Diabetes
0          66   59                285  ...              Yes           Yes      Yes

[1 rows x 9 columns]
Preprocessing new data... (The data is about to be squeaky clean!)

Predictions:
Random Forest Prediction: Heart Disease
Logistic Regression Prediction: No Heart Disease

Probabilities (Hey, at least computers are honest about their uncertainty!):
Random Forest Probabilities: [0.35 0.65]
Logistic Regression Probabilities: [0.50504645 0.49495355]
Creating a fancy radar chart... (No stealth technology required!)
Radar chart displayed! If you find yourself thinking, 'Hmm, this looks suspiciously scientific,' then our job here is done.

All done! If this suggests 'Heart Disease,' remember I'm just a bunch of code. Always see a real healthcare professional.
```