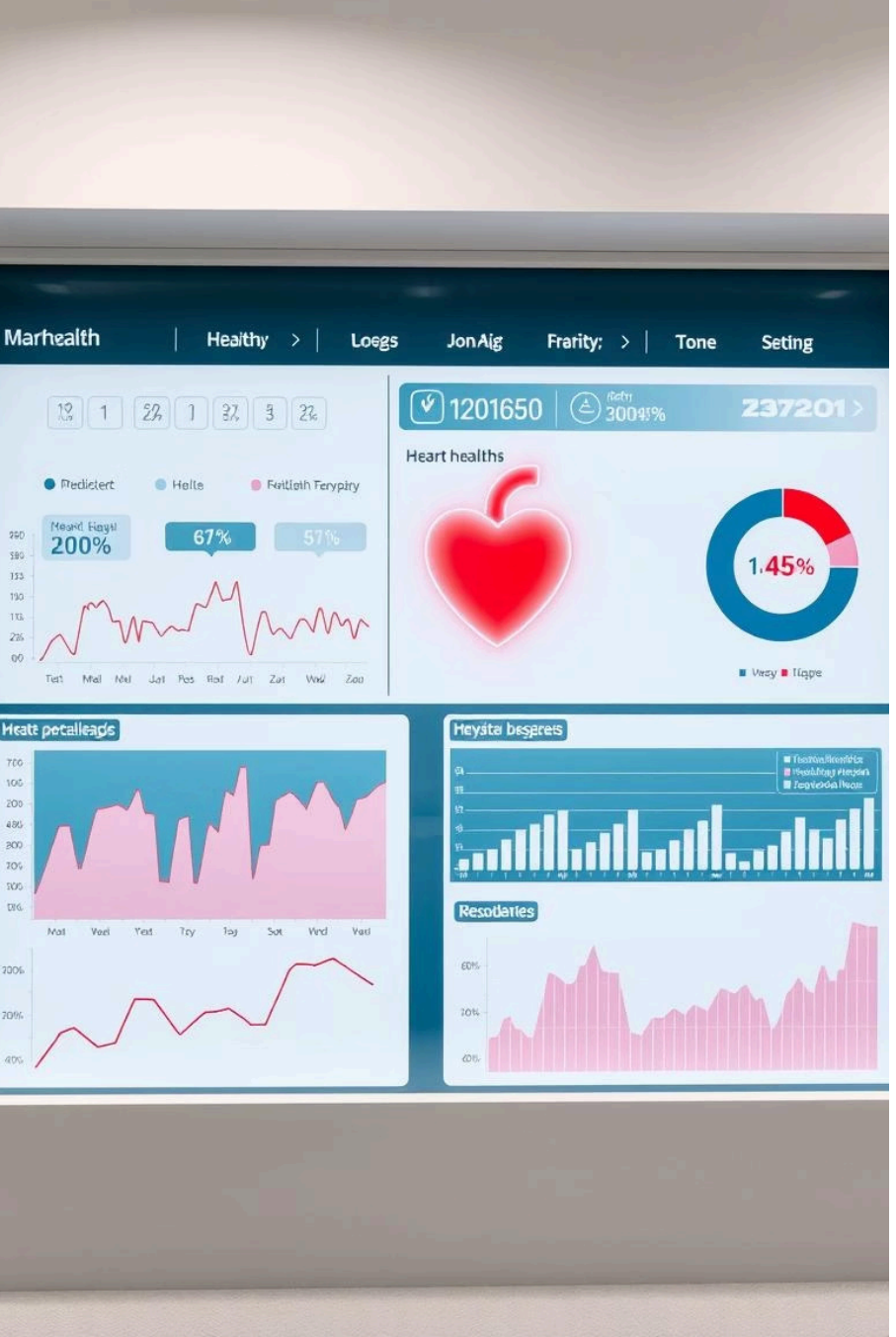


Heart Failure Prediction Using Classification

This project predicts heart failure risk using machine learning classification.

Project Workflow Overview

Data Collection

Load heart.csv dataset and inspect data.

Data Visualization

Explore distributions, correlations, and target balance.

Data Cleaning

Check for missing values and unique categories.

Data Transformation

Encode categorical features and scale numerical data.

Model Training and Evaluation

1 Initial Model

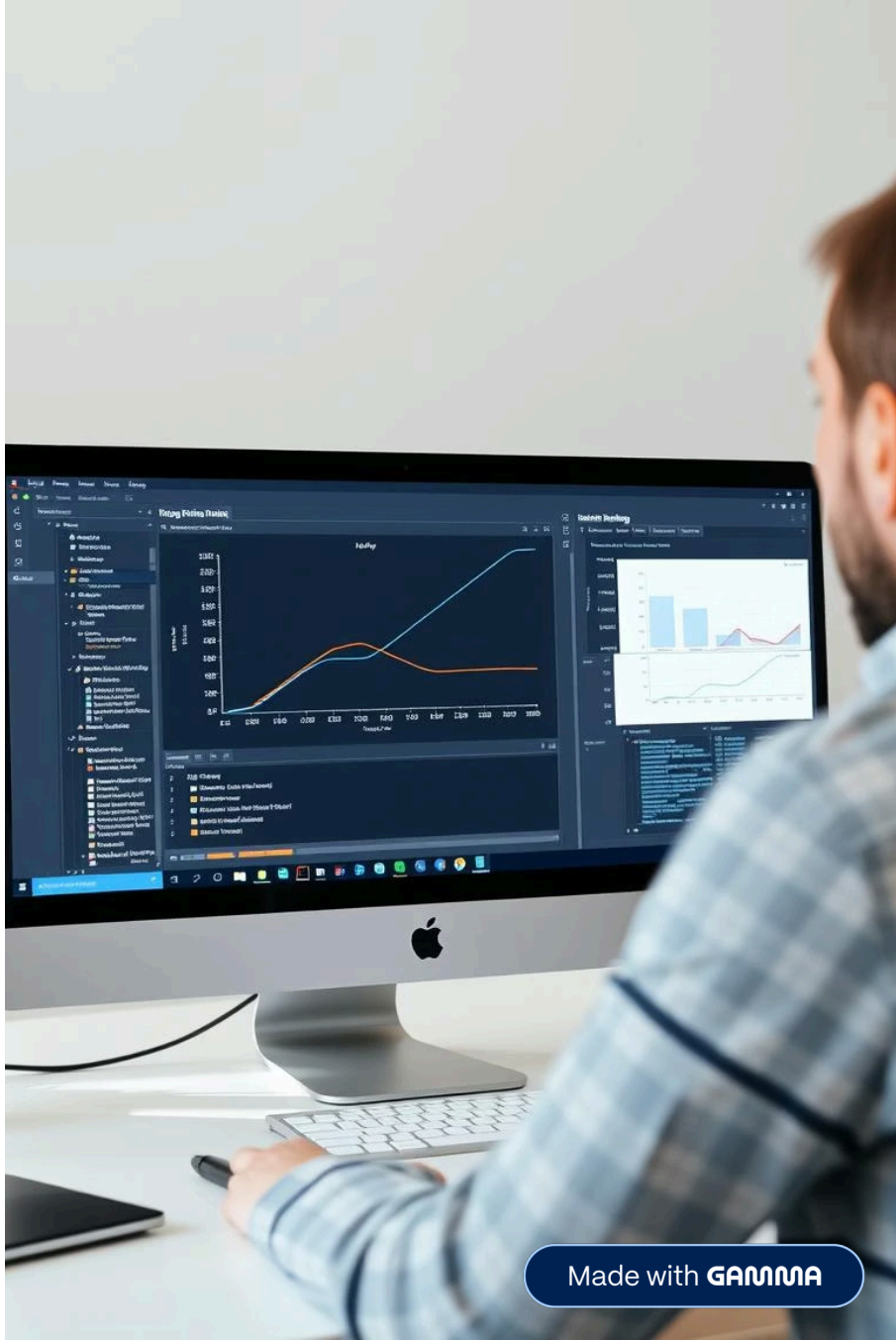
Train Logistic Regression baseline with 80/20 split.

2 Evaluation Metrics

Use accuracy, classification report, confusion matrix, ROC AUC.

3 Predictions

Assess model on test data for performance insights.



Hyperparameter Tuning with Random Forest

GridSearchCV

- Tune n_estimators, max_depth, min_samples_split, min_samples_leaf
- 5-fold cross-validation for accuracy
- Best estimator evaluated with classification report

RandomizedSearchCV

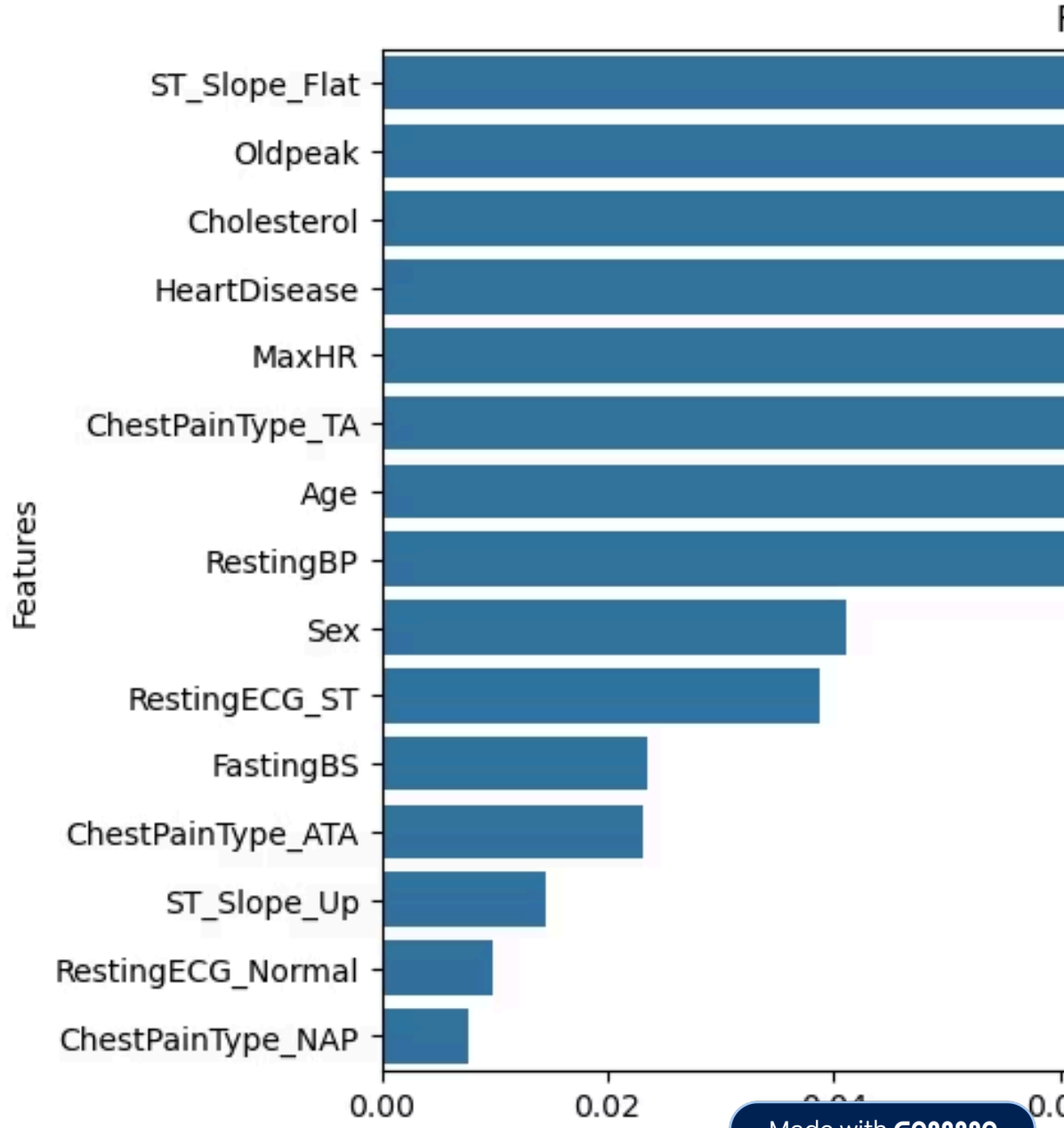
- Random hyperparameter sampling with 50 iterations
- Includes max_features tuning
- Best model assessed for accuracy and metrics

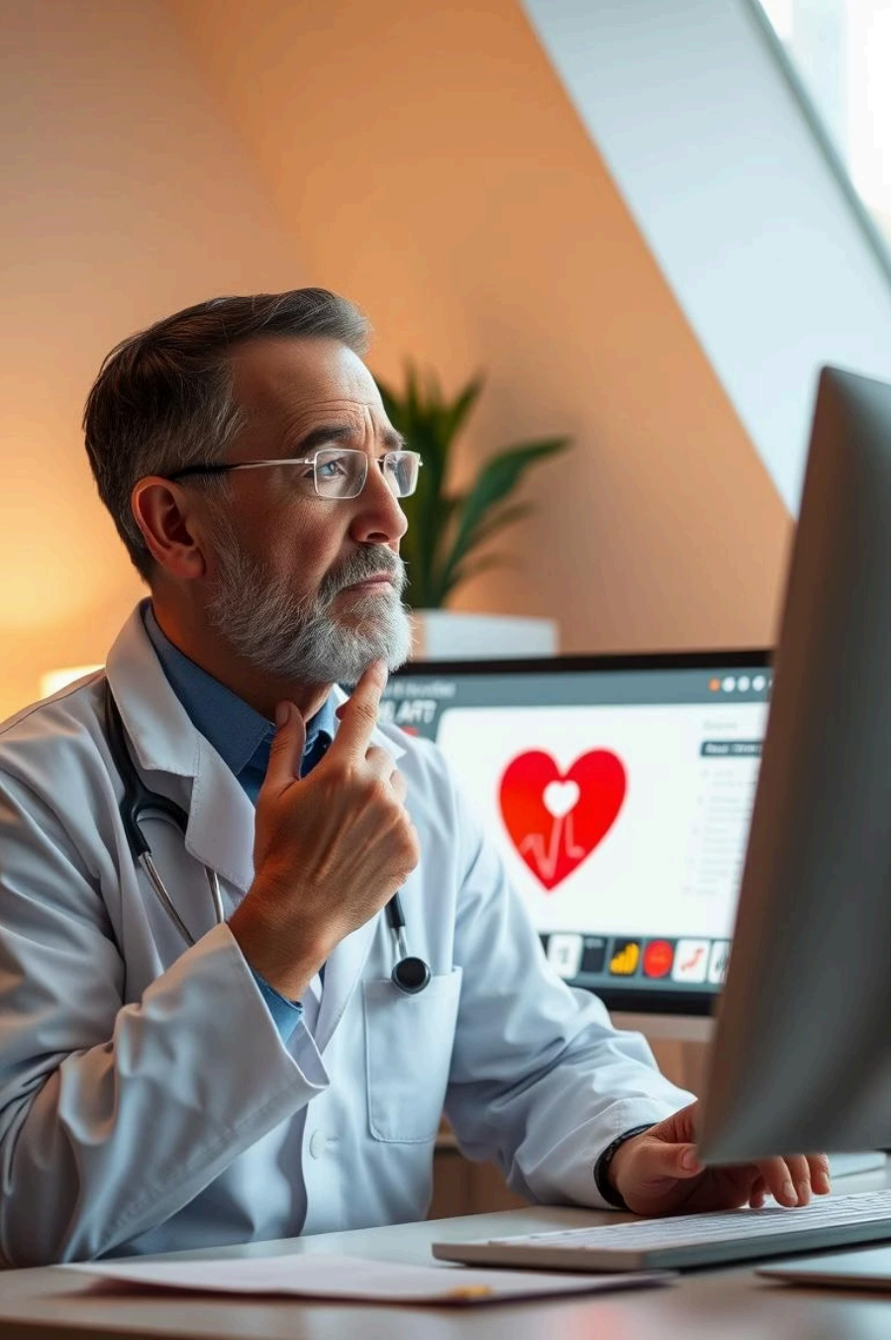
Model Performance Comparison

Model	Accuracy	Precision	Recall	F1	ROC AUC
Logistic Regression	0.85	0.86	0.85	0.85	0.93
Random Forest (GridSearchCV)	0.875	0.88	0.88	0.88	0.93
Random Forest (RandomizedSearchCV)	0.86	0.87	0.87	0.87	0.93

Feature Importance Analysis

The Random Forest model highlights key predictors of heart failure risk.





Project Conclusion

Successful Pipeline

Implemented full ML workflow for heart failure prediction.

Model Accuracy

Random Forest outperformed Logistic Regression baseline.

Insightful Features

Feature importance reveals critical health indicators.

How to Run the Project

1. Install Python and required libraries (pandas, scikit-learn, matplotlib, seaborn).
2. Place heart.csv in the notebook directory.
3. Open and run classification.ipynb in Jupyter environment.