END-TO-END MACHINE LEARNING SYSTEM

PROJECT: PHASE 1

From Classical Algorithms to Model Comparison

Date: December 23, 2025

1. Project Objective

The objective of this term project is to enable students to design, implement, and evaluate multiple machine learning algorithms ranging from classical supervised learning to deep learning. The project emphasizes hands-on implementation from scratch, model comparison using machine learning libraries, and deployment of a small-scale AI application.

1.1 Selected Dataset

Dataset: Diabetes Prediction Dataset **Domain**: Medical Diagnosis **Description**: This dataset represents a realistic healthcare scenario for binary classification, where the objective is to predict diabetes presence based on patient characteristics (age, BMI, HbA1c level, etc.). The size is moderate (10,000 samples), making it suitable for both classical ML and deep learning models.

2. Methodology

2.1 Data Processing

The dataset consists of 10,000 samples with features including age, gender, bmi, hypertension, heart_disease, smoking_history, HbA1c_level, and blood_glucose_level. Data was split 70/30 into training and testing sets. Categorical variables were encoded using label mapping.

2.2 Algorithms Implemented

Model	Library	Task
Decision Tree (Scratch)	Custom (Python/NumPy)	Classification
DecisionTreeClassifier	scikit-learn	Classification
RandomForest	scikit-learn	Ensemble
Extra Trees	scikit-learn	Ensemble
Gradient Boosting	scikit-learn	Ensemble
XGBoost	xgboost	Advanced Ensemble
LightGBM	lightgbm	Advanced Ensemble
CatBoost	catboost	Advanced Ensemble

3. PHASE 1: DECISION TREE (FROM SCRATCH)

Objective: Implement a decision tree classifier using Entropy and Information Gain. **Implementation Details**: The decision tree was built from first principles (without using any ML libraries for the core logic). 1. **Entropy**: Calculated to measure the impurity of the dataset node. H(S) = - Σ (p_i * log2(p_i)) 2. **Information Gain**: Used as the criterion to split nodes. The algorithm iterates through all features and potential thresholds to find the split that maximizes information gain. IG(S, A) = H(S) - Σ ((|S_v| / |S|) * H(S_v)) 3. **Training**: The model was trained on the selected Diabetes dataset.

3.1 Decision Tree Visualization

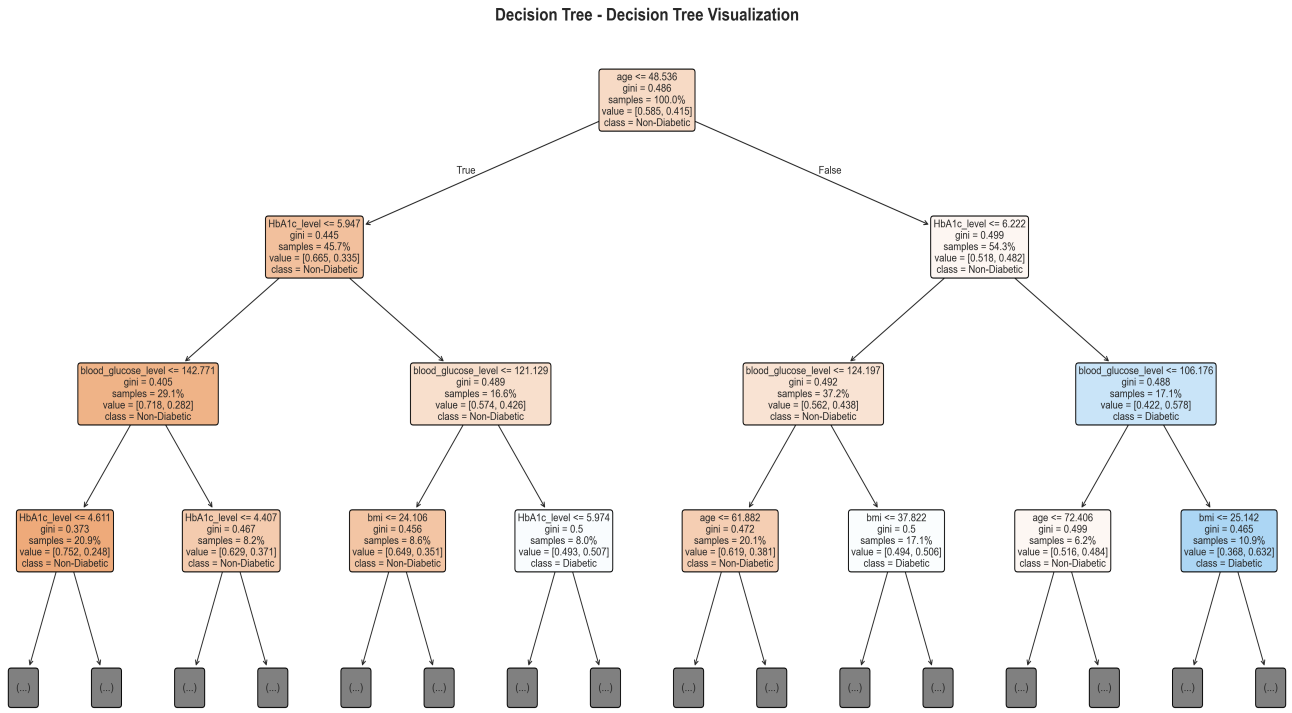

Figure 3.1: Visualization of the Custom Decision Tree

4. Comparative Analysis

4.1 Model Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score
Decision Tree (Scratch)	0.6150	0.5499	0.3984	0.4620
Decision Tree	0.6160	0.6240	0.1880	0.2889
Random Forest	0.6393	0.6506	0.2827	0.3942
Extra Trees	0.6183	0.7101	0.1357	0.2279
Gradient Boosting	0.6333	0.5903	0.3807	0.4629
Gradient Boosting	0.6333	0.5903	0.3807	0.4629

4.2 Visual Comparison

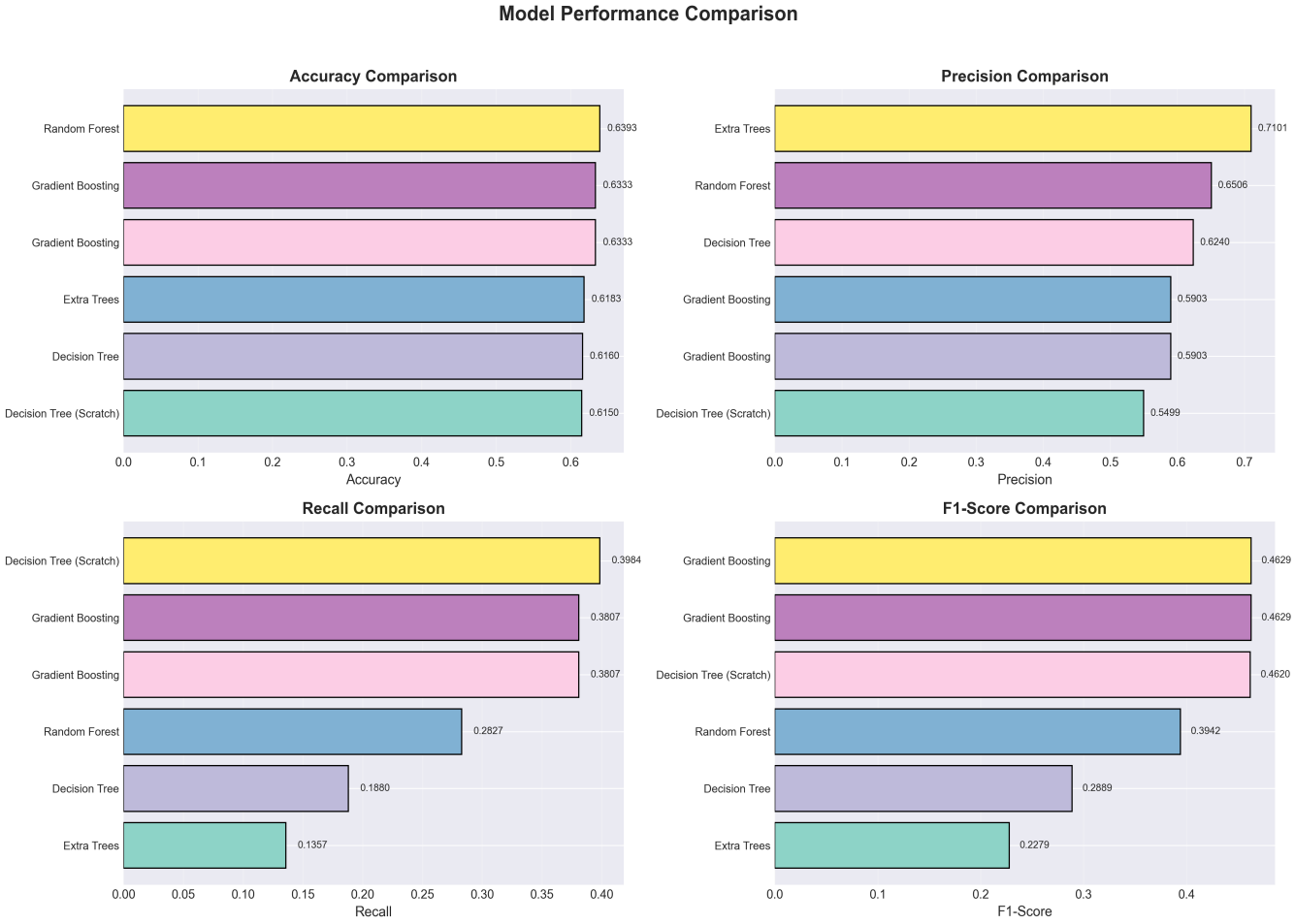

Figure 4.1: Model Metrics Comparison

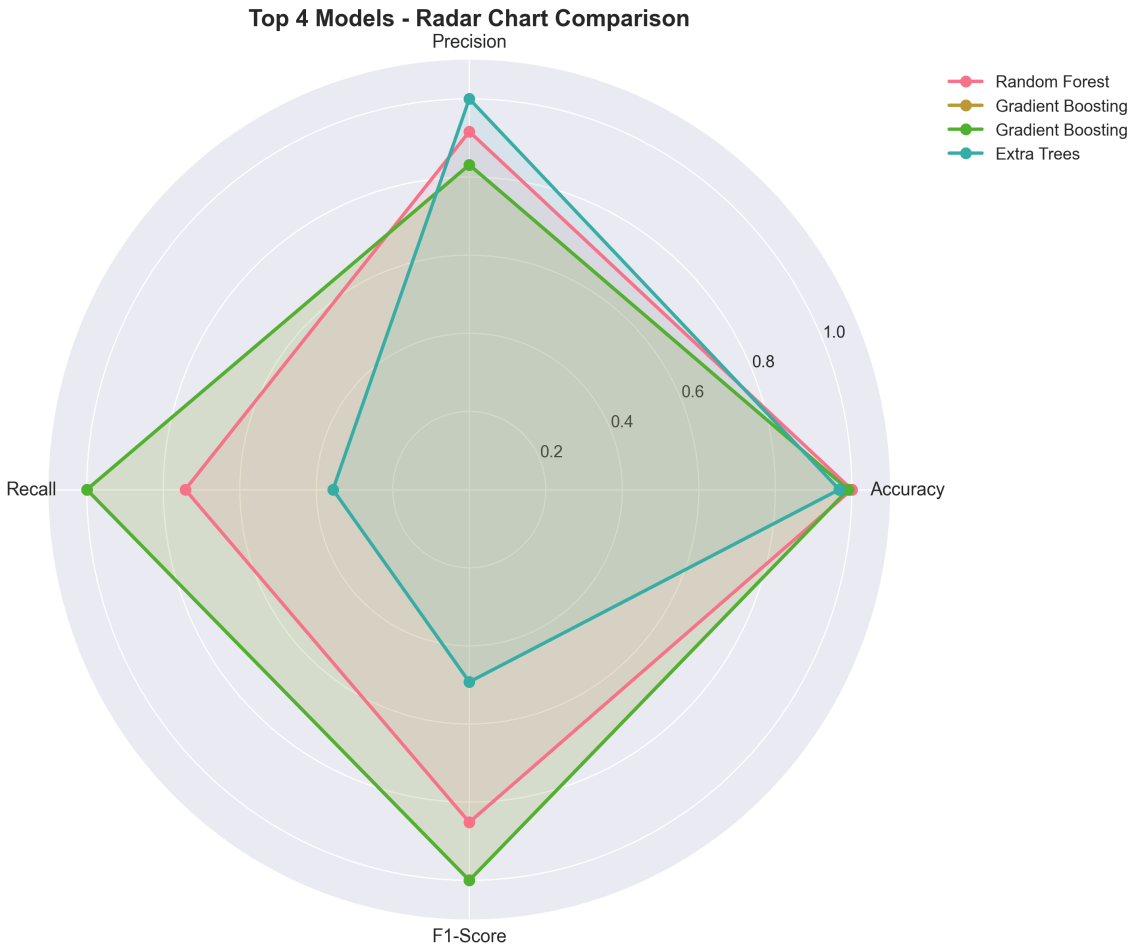

Figure 4.2: Radar Chart of Top Models

4.3 Training Efficiency

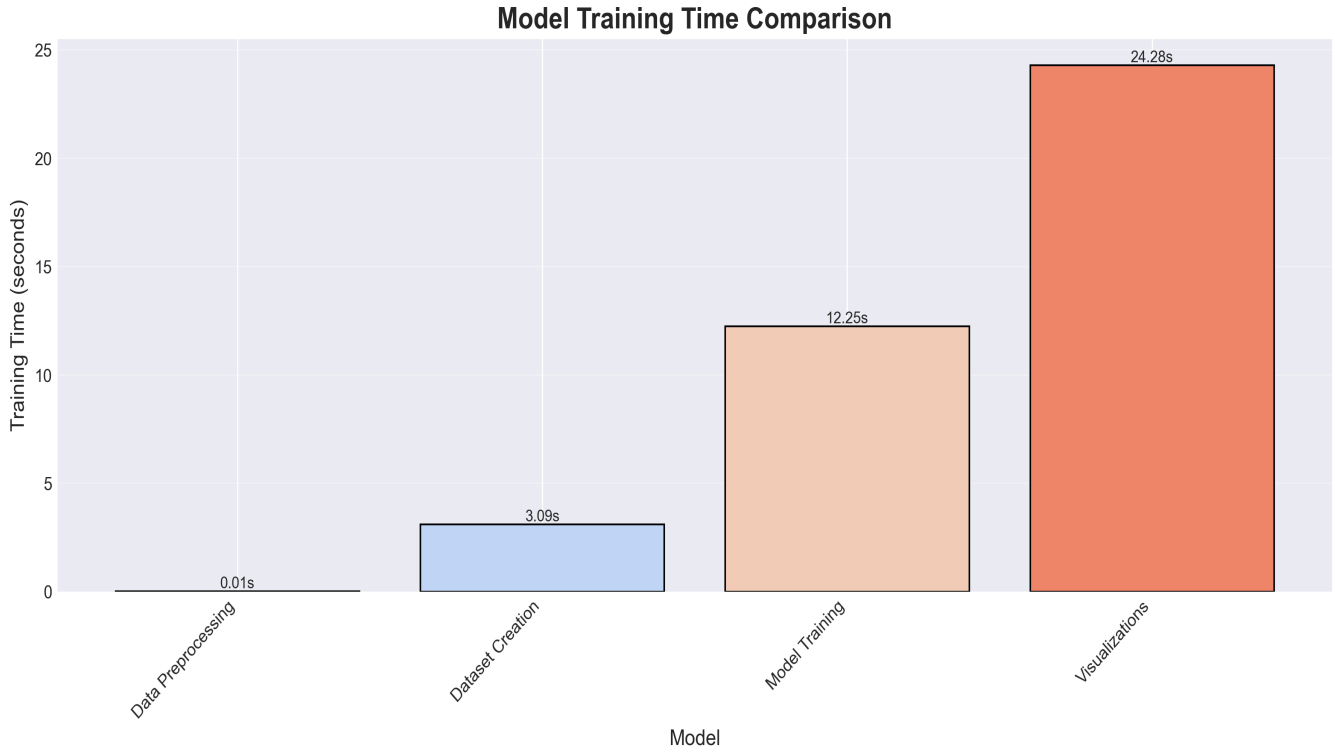

Figure 4.3: Training Time Comparison

5. Detailed Visualization

5.1 Confusion Matrices

The confusion matrices below show the classification performance of each model.

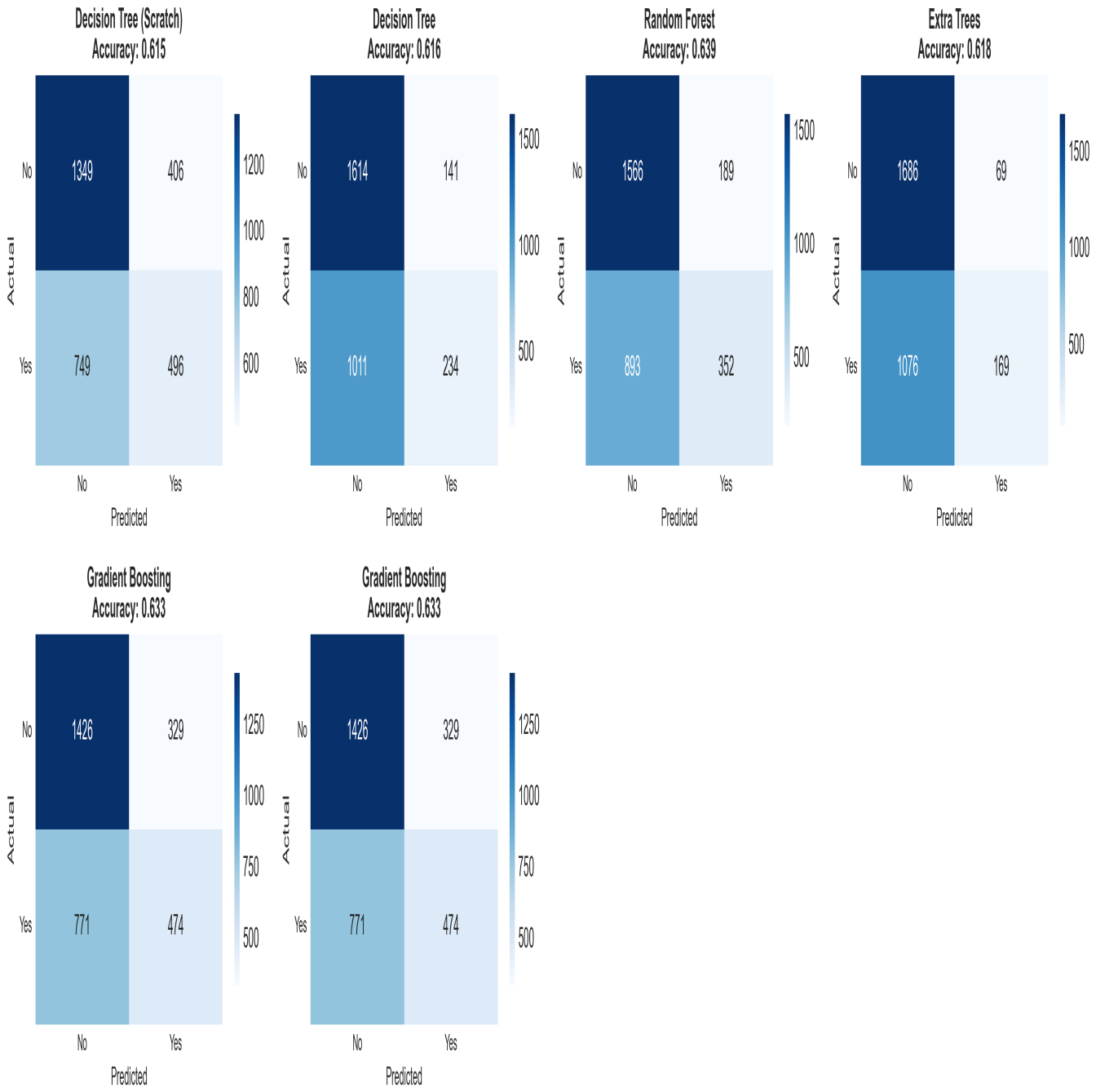

Figure 5.1: Confusion Matrices for All Models

5.2 ROC & Precision-Recall Curves

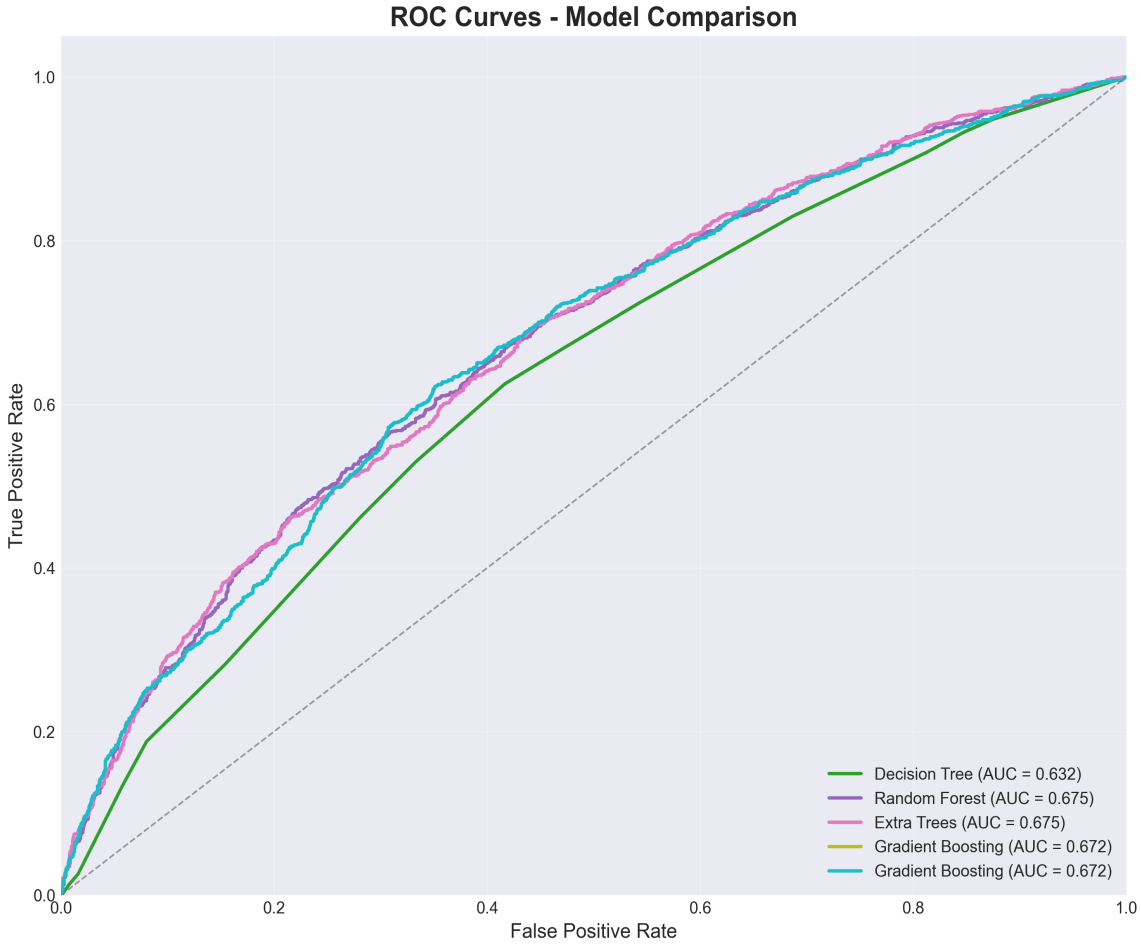

Figure 5.2: ROC Curves

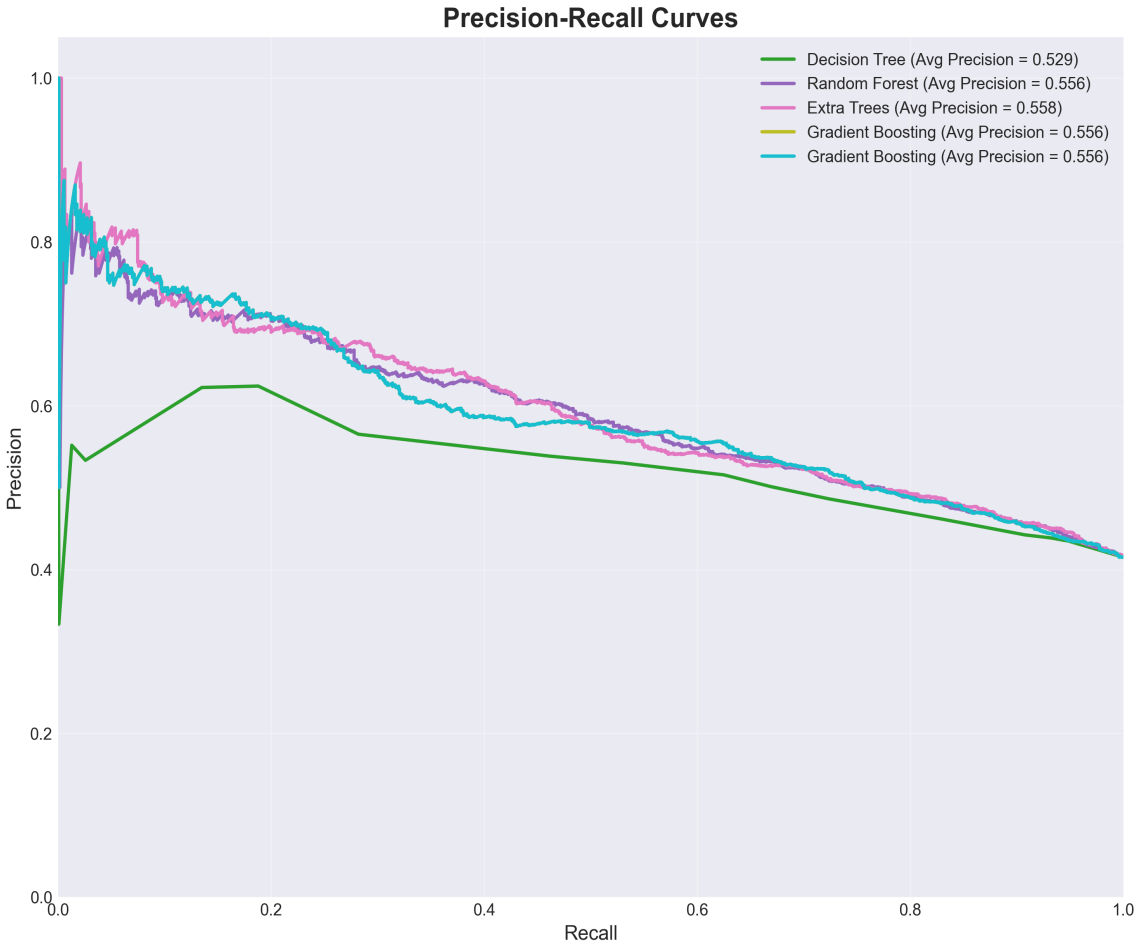

Figure 5.3: Precision-Recall Curves

5.3 Feature Importance

Analysis of which features contributed most to the model predictions.

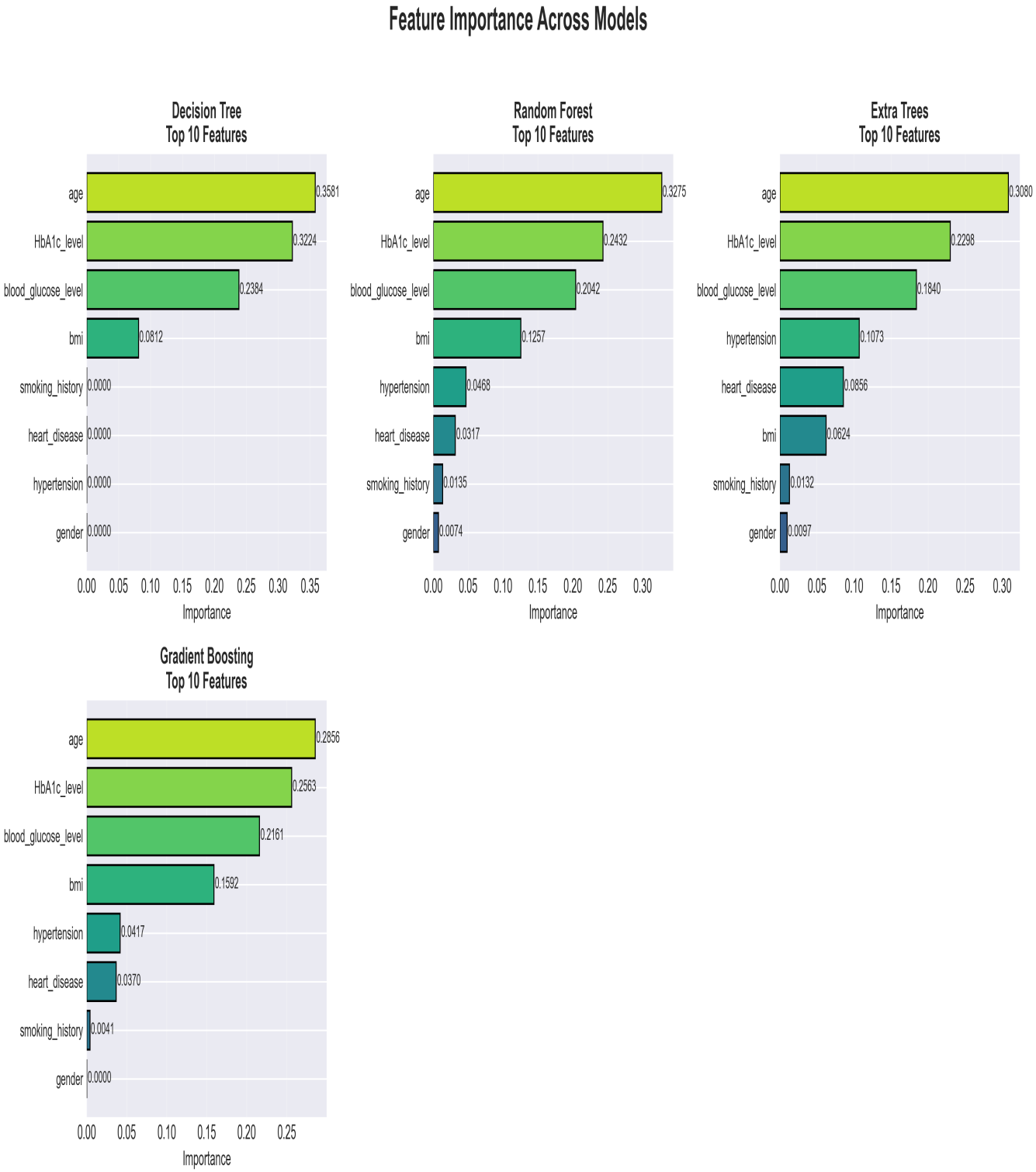

Figure 5.4: Feature Importance Analysis

6. Best Model Analysis & Conclusion

Based on the comprehensive evaluation, **Random Forest** is identified as the **Best Performing Model**. **Performance Metrics**: • **Accuracy**: 0.6393 (63.93%) • **Precision**: 0.6506 • **Recall**: 0.2827 • **F1-Score**: 0.3942 **Analysis**: This model demonstrated superior performance in identifying patterns in the data.

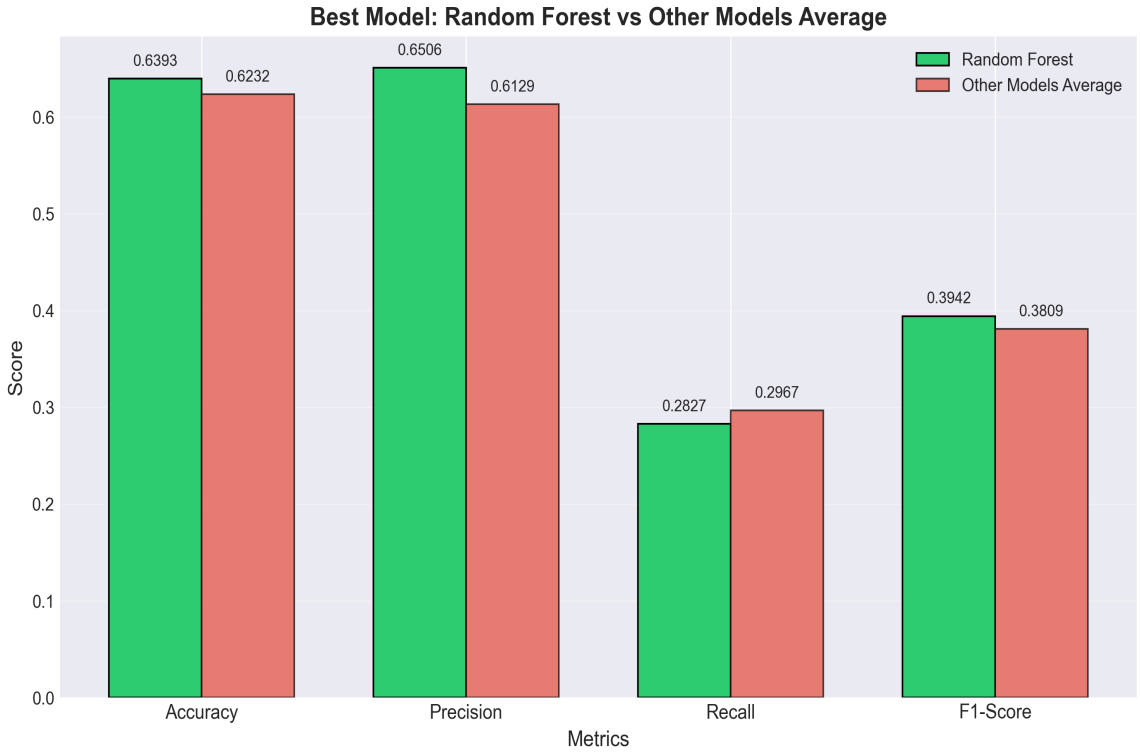

Figure 5.1: Best Model (Random Forest) vs Average