Automated Model Selection and Hyperparameter Optimization using Bayesian Optimization

# Overview

This document describes the implementation of an automated machine learning (AutoML) workflow for selecting the best model and tuning hyperparameters using Bayesian optimization via Optuna. The process includes training multiple models (Random Forest, SVC, and XGBoost), optimizing their parameters, and evaluating them using cross-validation.

# Dataset

The dataset used is the Breast Cancer Wisconsin dataset from `sklearn.datasets`. It contains features computed from breast mass images to classify tumors as malignant or benign.

# Preprocessing

Features are scaled using `StandardScaler` to normalize the input space, which improves the performance of most machine learning models, especially SVMs and gradient boosting methods.

# Optimization Method

Optuna is used for automated hyperparameter optimization. It employs Bayesian optimization through a technique called Tree-structured Parzen Estimator (TPE) to efficiently search for the optimal set of hyperparameters.

# Models Considered

The objective function dynamically selects among the following models:  
  
- Random Forest Classifier  
  
- Support Vector Classifier (SVC)  
  
- XGBoost Classifier  
  
Each model has a unique set of hyperparameters that are tuned during the optimization process.

# Objective Function

The Optuna objective function takes a trial and proposes a model type along with its corresponding hyperparameters. It evaluates the model using 3-fold cross-validation and returns the mean accuracy as the optimization metric.

## Example of Hyperparameters Tuned

- Random Forest: `n\_estimators`, `max\_depth`  
  
- SVC: `C`, `gamma`  
  
- XGBoost: `n\_estimators`, `learning\_rate`, `max\_depth`

# Cross-Validation

The workflow uses 3-fold cross-validation to ensure generalizability and robustness of the model performance metrics.

# Deployment Readiness

This setup is ideal for deployment as it can:  
  
- Automatically determine the best-performing model  
  
- Tune it with optimal hyperparameters  
  
- Easily integrate into a pipeline for real-time or batch predictions

# Tools and Libraries Used

- scikit-learn  
- Optuna  
- XGBoost  
- pandas, numpy  
- StandardScaler for preprocessing