Model Optimization and Tuning Phase Report

Date	10s July 2024
Team ID	739835
Project Title	Credit card approval prediction using ML
Maximum Marks	10 Marks

Model Optimization and Tuning Phase

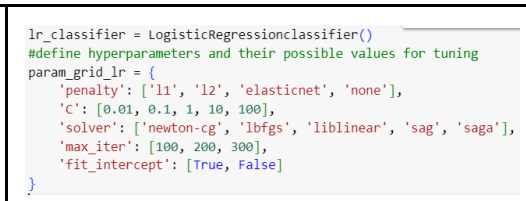
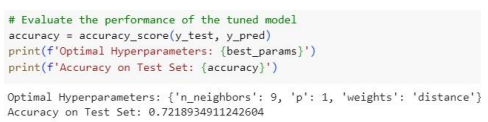
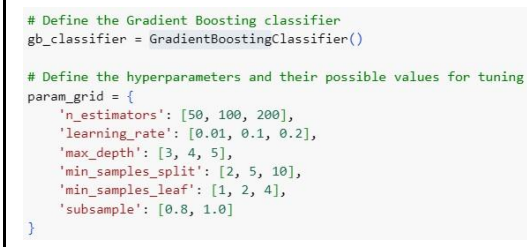
The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

Hyperparameter Tuning Documentation (6 Marks):

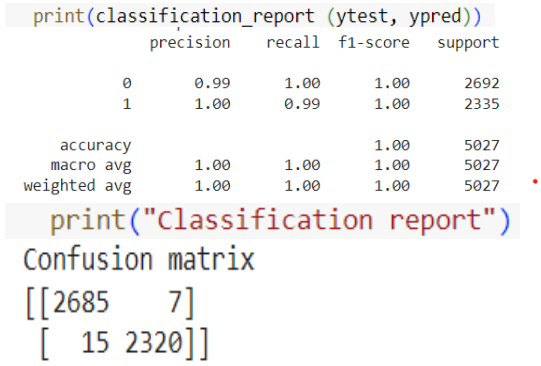
Model	Tuned Hyperparameters	Optimal Values

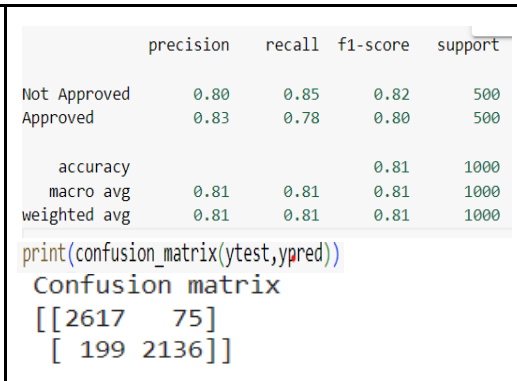
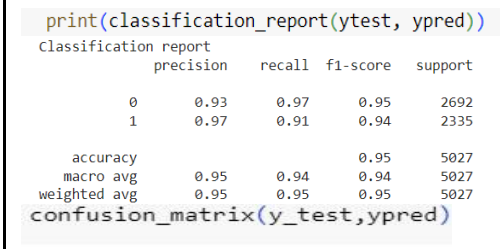
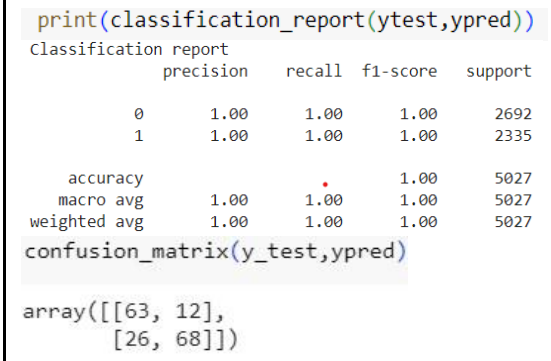
Decision Tree	`# Define the Decision Tree classifier`
	`dt_classifier = DecisionTreeClassifier()`
	
	`# Define the hyperparameters and their possible values for tuning`
	`param_grid = {`
	`'criterion': ['gini', 'entropy'],`
	`'splitter': ['best', 'random'],`
	`'max_depth': [None, 10, 20, 30, 40, 50],`
	`'min_samples_split': [2, 5, 10],`
	`'min_samples_leaf': [1, 2, 4]`
	`}`
		`# Evaluate the performance of the tuned model`
		`accuracy = accuracy_score(y_test, y_pred)`
		`print(f'Optimal Hyperparameters: {best_params}')`
		`print(f'Accuracy on Test Set: {accuracy}')`
		
		`Optimal Hyperparameters: {'criterion': 'gini', 'max_depth': None, 'min_samples_leaf': 2, 'min_samples_split': 10, 'splitter': 'best'}`
		`Accuracy on Test Set: 0.7159763313609467`

Random Forest	`# Define the Random Forest classifier`
	`rf_classifier = RandomForestClassifier()`
	
	`# Define the hyperparameters and their possible values for tuning`
	`param_grid = {`
	`'n_estimators': [50, 100, 200],`
	`'criterion': ['gini', 'entropy'],`
	`'max_depth': [None, 10, 20, 30],`
	`'min_samples_split': [2, 5, 10],`
	`'min_samples_leaf': [1, 2, 4],`
	`}`
		`# Evaluate the performance of the tuned model`
		`accuracy = accuracy_score(y_test, y_pred)`
		`print(f'Optimal Hyperparameters: {best_params}')`
		`print(f'Accuracy on Test Set: {accuracy}')`
		
		`Optimal Hyperparameters: {'criterion': 'entropy', 'max_depth': 20, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100}`
		`Accuracy on Test Set: 0.7751479289940828`

Logistic Regression		
Gradient Boosting		

Performance Metrics Comparison Report (2 Marks):

Model	Optimized Metric
Decision Tree	

Random Forest	
Logistic Regression	
Gradient Boosting	

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
Gradient Boosting	The Gradient Boosting model was selected for its superior performance, exhibiting high accuracy during hyperparameter tuning. Its ability to handle complex relationships, minimize overfitting, and optimize predictive accuracy aligns with project objectives, justifying its selection as the final model.