

Model Optimization and Tuning Phase Report

Date	15 July 2024
Team ID	739713
Project Title	Beansense: Precision Bean Classification For Enhanced Agricultural And Culinary Applications
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.

Model	Tuned Hyperparameters	Optimal Values
Decision Tree	<pre>[] from sklearn.tree import DecisionTreeClassifier</pre> <pre>[] model2=DecisionTreeClassifier()</pre> <pre>▶ model2.fit(xtrain,ytrain)</pre> <pre>↔ DecisionTreeClassifier</pre> <pre>DecisionTreeClassifier()</pre> <pre>[] ypred=model2.predict(xtest)</pre>	<pre>accuracy2=accuracy_score(ytest,ypred)</pre> <pre>precision2=precision_score(ytest,ypred,average="weighted")</pre> <pre>recall2=recall_score(ytest,ypred,average="weighted")</pre> <pre>conf2=confusion_matrix(ytest,ypred)</pre> <pre>classrep2=classification_report(ytest,ypred)</pre> <pre>accuracy2,precision2,recall2,conf2,classrep2</pre>

<p>Random Forest</p>	<pre>from sklearn.ensemble import RandomForestClassifier model3 = RandomForestClassifier() model3.fit(xtrain,ytrain)</pre> <div> <div>▼ RandomForestClassifier</div> <div>RandomForestClassifier()</div> </div>	<pre>from sklearn.metrics import accuracy_score,precision_score ypred = model3.predict(xtest) accuracy = accuracy_score(ytest, ypred) precision = precision_score(ytest, ypred,average='weighted') recall = recall_score(ytest, ypred,average="weighted") f1 = f1_score(ytest, ypred,average="weighted") confusion = confusion_matrix(ytest, ypred) print("Accuracy:", accuracy) print("Precision:", precision) print("Recall:", recall) print("F1 Score:", f1) print("Confusion Matrix:\n", confusion)</pre>
<p>Hyperparameter Tuning</p>	<pre>from sklearn.model_selection import GridSearchCV from sklearn.ensemble import RandomForestClassifier</pre>	<pre>param_grid = { 'n_estimators': [100, 200, 300], 'max_depth': [10, 20, 30, None], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'max_features': ['auto', 'sqrt', 'log2'] } param_dist = { 'n_estimators': [100, 200, 300, 400, 500], 'max_depth': [10, 20, 30, 40, 50, None], 'min_samples_split': [2, 5, 10, 20], 'min_samples_leaf': [1, 2, 4, 8], 'max_features': ['auto', 'sqrt', 'log2'] }</pre>

Performance Metrics Comparison Report (2 Marks):

Model	Optimized Metric																																								
Decision Tree	<div><pre>(0.87862368867414, 0.8789976418676767, 0.87862368867414, array([[237, 24, 0, 2, 3, 5], [26, 277, 0, 11, 1, 4], [0, 0, 681, 5, 9, 63], [7, 18, 4, 314, 0, 9], [2, 0, 25, 0, 372, 12], [6, 0, 58, 15, 12, 412]])</pre></div> <table><tr><td>ROC</td><td>0.91</td><td>0.91</td><td>0.91</td><td>308%</td><td>SIXER</td><td>0.94</td><td>0.91</td><td>0.92</td><td>400%</td><td>SIRA</td><td>0.83</td><td>0.81</td><td>0.83</td><td>543%</td><td>accuracy</td><td>0.88</td><td>0.89</td><td>0.88</td><td>678%</td></tr><tr><td>0.88</td><td>0.88</td><td>0.88</td><td>2587%</td><td>unweighted avg</td><td>0.88</td><td>0.88</td><td>0.88</td><td>2587%</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>2587%</td><td>macro avg</td><td></td><td></td></tr></table>	ROC	0.91	0.91	0.91	308%	SIXER	0.94	0.91	0.92	400%	SIRA	0.83	0.81	0.83	543%	accuracy	0.88	0.89	0.88	678%	0.88	0.88	0.88	2587%	unweighted avg	0.88	0.88	0.88	2587%								2587%	macro avg		
ROC	0.91	0.91	0.91	308%	SIXER	0.94	0.91	0.92	400%	SIRA	0.83	0.81	0.83	543%	accuracy	0.88	0.89	0.88	678%																						
0.88	0.88	0.88	2587%	unweighted avg	0.88	0.88	0.88	2587%								2587%	macro avg																								

Random Forest	<p>Accuracy: 0.9130266718206417 Precision: 0.9133037264533918 Recall: 0.9130266718206417 F1 Score: 0.9130395511643149 Confusion Matrix:</p> <pre>[[245 19 0 1 2 4] [14 291 0 9 1 4] [0 0 630 1 5 42] [2 12 2 347 0 5] [2 0 16 0 380 10] [3 3 51 10 7 469]]</pre>
Hyperparameter Tuning	 <pre>Tuned Model Performance: Accuracy: 0.9222737819025522 Precision: 0.9226830175848943 Recall: 0.9222737819025522 F1 Score: 0.9221833236192046 Confusion Matrix: [[238 25 0 1 2 6] [9 287 0 7 2 3] [0 0 647 1 6 34] [3 1 1 372 0 6] [2 0 16 0 380 10] [1 1 52 8 4 461]]</pre>

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
Classifier	Classifiers in machine learning are algorithms that learn to classify data into predefined categories or classes based on labeled training data. They enable tasks like spam detection, image recognition, and sentiment analysis by predicting the class of new, unseen data points.