

Model Optimization and Tuning Phase Template

Date	2 nd July 2024
Team ID	SWTID1720109498
Project Title	Blueberry Yield Predictor
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
Linear Regression	-	<p>Regression metrics on the test set</p> <p>R2 score: 0.9850297685259484</p> <p>MAE: 111.5204986497179</p> <p>MSE: 21684.627147497344</p> <p>RMSE: 147.25701052071287</p>
Random Forest Regressor	<pre>from sklearn.model_selection import GridSearchCV param_grid = { 'n_estimators': [100, 200, 300], 'max_features': ['auto', 'sqrt', 'log2'], 'max_depth': [10, 20, 30, None], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'bootstrap': [True, False] } rf = RandomForestRegressor() grid_search = GridSearchCV(estimator=rf, param_grid=param_grid, cv=3, n_jobs=-1, verbose=2)</pre>	<p>Fitting 3 folds for each of 648 candidates, totalling 1944 fits</p> <p>Best Parameters: {'bootstrap': False, 'max_depth': None, 'max_features': 'log2', 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 200}</p> <p>Mean Squared Error: 23036.512805030303</p> <p>R-squared: 0.9840964707311984</p>

DecisionTreeRegressor	<pre>param_grid = { 'max_depth': [None, 10, 20, 30], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'max_features': [None, 'auto', 'sqrt', 'log2'] } dt = DecisionTreeRegressor() grid_search = GridSearchCV(estimator=dt, param_grid=param_grid, cv=3, n_jobs=-1, verbose=2)</pre>	<p>Fitting 3 folds for each of 144 candidates, totalling 432 fits</p> <p>Best Parameters: {'max_depth': 30, 'max_features': None, 'min_samples_leaf': 4, 'min_samples_split': 2}</p> <p>Mean Squared Error: 39945.42992962877</p> <p>R-squared: 0.9724232135369657</p>
XGBRegressor	<pre>param_grid = { 'n_estimators': [100, 200, 300], 'learning_rate': [0.01, 0.1, 0.2], 'max_depth': [3, 6, 9], 'subsample': [0.6, 0.8, 1.0], 'colsample_bytree': [0.6, 0.8, 1.0] } xgb = XGBRegressor() grid_search = GridSearchCV(estimator=xgb, param_grid=param_grid, cv=3, n_jobs=-1, verbose=2)</pre>	<p>Fitting 3 folds for each of 243 candidates, totalling 729 fits</p> <p>Best Parameters: {'colsample_bytree': 1.0, 'learning_rate': 0.1, 'max_depth': 3, 'n_estimators': 200, 'subsample': 0.6}</p> <p>Mean Squared Error: 19206.509150671092</p> <p>R-squared: 0.9867405657547946</p>

Performance Metrics Comparison Report (2 Marks):

Model	Baseline Metric	Optimized Metric
Linear Regression	<p>Regression metrics on the test set</p> <p>R2 score: 0.9850297685259484</p> <p>MAE: 111.5204986497179</p> <p>MSE: 21684.627147497344</p> <p>RMSE: 147.25701052071287</p>	-
Random Forest Regressor	<p>Regression metrics on the test set</p> <p>R2 score: 0.9794220417457439</p> <p>MAE: 132.1608756852944</p> <p>MSE: 29807.511859370235</p> <p>RMSE: 172.64852116183977</p>	<p>Regression metrics on the test set</p> <p>R2 score: 0.9840964787311984</p> <p>MAE: 120.52941809457525</p> <p>MSE: 23036.512805030303</p> <p>RMSE: 151.7778402963697</p>
Decision Tree Regressor	<p>Regression metrics on the test set</p> <p>R2 score: 0.9621834493005946</p> <p>MAE: 181.75958724836605</p> <p>MSE: 54777.89727850463</p> <p>RMSE: 234.04678437975736</p>	<p>Regression metrics on the test set</p> <p>R2 score: 0.9724232135369657</p> <p>MAE: 155.86761325904146</p> <p>MSE: 39945.42992962877</p> <p>RMSE: 199.86352826273423</p>
XGB Regressor	<p>Regression metrics on the test set</p> <p>R2 score: 0.982172684010463</p> <p>MAE: 119.19604783486517</p> <p>MSE: 25823.161181161377</p> <p>RMSE: 160.69586547625107</p>	<p>Regression metrics on the test set</p> <p>R2 score: 0.9867405657547946</p> <p>MAE: 108.15756897906451</p> <p>MSE: 19206.509150671092</p> <p>RMSE: 138.5875504894689</p>

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
XGB Regressor	The XGB Regressor model was selected for its superior performance, exhibiting the highest R2 score 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