Model Development Phase Template

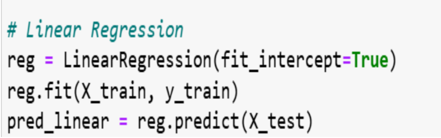
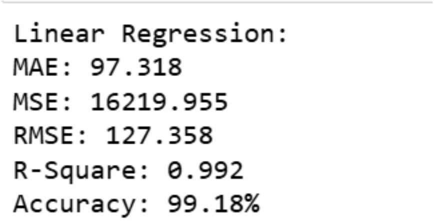
Date	20 July 2024
Team ID	SWTID1721319573
Project Title	Blueberry Yield Prediction
Maximum Marks	6 Marks

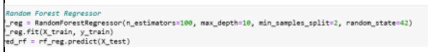
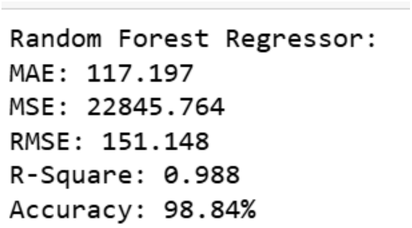
Model Selection Report

In the forthcoming Model Selection Report, various models will be outlined, detailing their descriptions, hyperparameters, and performance metrics, including Accuracy or F1 Score. This comprehensive report will provide insights into the chosen models and their effectiveness.

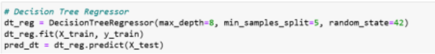
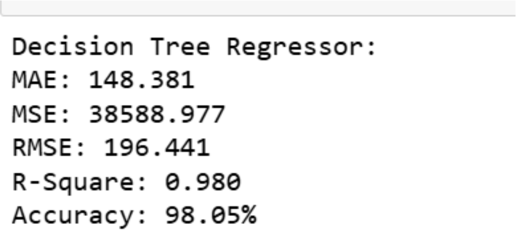
Model Selection Report:

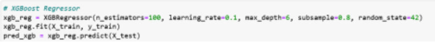
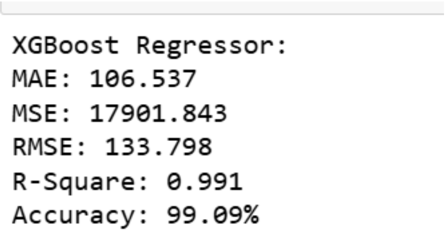
Model	Description	Hyperparameters	Performance Metric (e.g., Accuracy, F1 Score)

Linear Regression	Linear Regression is a simple model that assumes a linear relationship between the independent variables and the dependent variable.	
		# Linear Regression
		reg = LinearRegression(fit_intercept=True)
		reg.fit(X_train, y_train)
		pred_linear = reg.predict(X_test)
			
			Linear Regression:
			MAE: 97.318
			MSE: 16219.955
			RMSE: 127.358
			R-Square: 0.992
			Accuracy: 99.18%

Random Forest Regressor	Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the mean prediction	
		Random Forest Regressor
		_reg = RandomForestRegressor(n_estimators=100, max_depth=10, min_samples_split=2, random_state=42)
		_reg.fit(X_train, y_train)
		ed_rf = rf_reg.predict(X_test)
			
			Random Forest Regressor:
			MAE: 117.197
			MSE: 22845.764
			RMSE: 151.148
			R-Square: 0.988
			Accuracy: 98.84%

	of the individual trees.		

Decision Tree Regressor	Decision Tree Regressor creates a model in the form of a tree structure, where each node represents a decision based on the features, and the leaves represent the predicted values.	
		`# Decision Tree Regressor`
		`dt_reg = DecisionTreeRegressor(max_depth=8, min_samples_split=5, random_state=42)`
		`dt_reg.fit(X_train, y_train)`
		`pred_dt = dt_reg.predict(X_test)`
			
			`Decision Tree Regressor:`
			`MAE: 148.381`
			`MSE: 38588.977`
			`RMSE: 196.441`
			`R-Square: 0.980`
			`Accuracy: 98.05%`

XGBoost Regressor	XGBoost is an advanced gradient boosting method that optimizes the performance of boosting algorithms and is known for its accuracy and efficiency.	
		`# XGBoost Regressor`
		`xgb_reg = XGBRegressor(n_estimators=100, learning_rate=0.1, max_depth=6, subsample=0.8, random_state=42)`
		`xgb_reg.fit(X_train, y_train)`
		`pred_xgb = xgb_reg.predict(X_test)`
			
			`XGBoost Regressor:`
			`MAE: 106.537`
			`MSE: 17901.843`
			`RMSE: 133.798`
			`R-Square: 0.991`
			`Accuracy: 99.09%`