

Model Development Phase

Date	06 July 2024
Team ID	739915
Project Title	BlueBerry Yield Prediction
Maximum Marks	4 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include a summary and training and validation performance metrics for multiple models, presented through respective screenshots.

Initial Model Training Code :

```
[36]: lr = LinearRegression()  
      lr.fit(x_train,y_train)  
      pred_lr=lr.predict(x_test)
```

```
rf=RandomForestRegressor(max_depth=1)  
rf.fit(x_train,y_train)  
pred_rf=rf.predict(x_test)  
pred_rf_train=rf.predict(x_train)
```

```
dt=DecisionTreeRegressor()  
dt.fit(x_train,y_train)  
pred_dt=dt.predict(x_test)
```

```
xgb=XGBRegressor()  
xgb.fit(x_train,y_train)  
pred_xgb=xgb.predict(x_test)
```

Model Validation and Evaluation Report :

Model	Summary	Training and Validation Performance Metrics
Linear Regression	<pre> mae_lr = mean_absolute_error(y_test,pred_lr) mse_lr = mean_squared_error(y_test,pred_lr) rmse_lr = np.sqrt(mse_lr) rsq_lr = r2_score(y_test,pred_lr) print("MAE:%.3f" % mae_lr) print("MSE:%.3f" % mse_lr) print("RSME:%.3f" % rmse_lr) print("R-Square:%.3f" % rsq_lr) print("training accuracy",lr.score(x_train,y_train)) print("testing accuracy",lr.score(x_test,y_test)) </pre>	<p>MAE:87.009 MSE:12093.240 RSME:109.969 R-Square:0.993 training accuracy 0.9918401736922372 testing accuracy 0.992515823698853</p>
Random Forest	<pre> mae_rf_train=mean_absolute_error(y_train,pred_rf_train) mae_rf = mean_absolute_error(y_test,pred_rf) mse_rf = mean_squared_error(y_test,pred_rf) rmse_rf = np.sqrt(mse_rf) rsq_rf = r2_score(y_test,pred_rf) print("MAE_train:%.3f" % mae_rf_train) print("MAE:%.3f" % mae_rf) print("MSE:%.3f" % mse_rf) print("RSME:%.3f" % rmse_rf) print("R-Square:%.3f" % rsq_rf) print("training accuracy",lr.score(x_train,y_train)) print("testing accuracy",lr.score(x_test,y_test)) </pre>	<p>MAE_train:590.559 MAE:590.240 MSE:484111.690 RSME:695.781 R-Square:0.700 training accuracy 0.9918401736922372 testing accuracy 0.992515823698853</p>
Decision Tree	<pre> mae_dt = mean_absolute_error(y_test,pred_dt) mse_dt = mean_squared_error(y_test,pred_dt) rmse_dt = np.sqrt(mse_dt) rsq_dt = r2_score(y_test,pred_dt) print("MAE:%.3f" % mae_dt) print("MSE:%.3f" % mse_dt) print("RSME:%.3f" % rmse_dt) print("R-Square:%.3f" % rsq_dt) print("training accuracy",dt.score(x_train,y_train)) print("testing accuracy",dt.score(x_test,y_test)) </pre>	<p>MAE:159.751 MSE:42218.450 RSME:205.471 R-Square:0.974 training accuracy 1.0 testing accuracy 0.9738721523374747</p>
XGBoost	<pre> mae_xgb = mean_absolute_error(y_test,pred_xgb) mse_xgb = mean_squared_error(y_test,pred_xgb) rmse_xgb = np.sqrt(mse_xgb) rsq_xgb = r2_score(y_test,pred_xgb) print("MAE:%.3f" % mae_xgb) print("MSE:%.3f" % mse_xgb) print("RSME:%.3f" % rmse_xgb) print("R-Square:%.3f" % rsq_xgb) print("training accuracy",xgb.score(x_train,y_train)) print("testing accuracy",xgb.score(x_test,y_test)) </pre>	<p>MAE:107.235 MSE:19564.571 RSME:139.873 R-Square:0.988 training accuracy 0.9999402183903177 testing accuracy 0.9878920205537743</p>