

Project Development Phase  
Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID592078
Project Name	Project - Predicting Lumpy Skin Disease
Maximum Marks	10 Marks

Model Performance Testing:

S.No.	Parameter	Values	Screenshot
1.	Metrics	<p><b>Regression Model:</b> MAE - , MSE - , RMSE - , R2 score -</p> <p><b>Classification Model:</b> Confusion Matrix - , Accuray Score- &amp; Classification Report -</p>	<p><b>Linear Regression Model:</b></p> <pre>Activity 2.1: Logistic Regression Model</pre> <pre>In [34]: from sklearn.linear_model import LogisticRegression In [35]: log = LogisticRegression() In [36]: log.fit(X_train, y_train) In [37]: y_pred = log.predict(X_test) In [38]: from sklearn.metrics import accuracy_score, confusion_matrix, classification_report, roc_auc_score, roc_curve In [39]: accuracy_score(y_test, y_pred) Out[39]: 0.8571428571428571 In [40]: confusion_matrix(y_test, y_pred) Out[40]: array([[10, 0],               [ 0, 10]]) In [41]: accuracy_score(y_test, y_pred) Out[41]: 1.0 In [42]: print(classification_report(y_test, y_pred)) precision    recall  f1 score   support 0.00         0.00         0.00         10 1.00         1.00         1.00         10 avg / total         0.86         0.86         0.86         20</pre> <p><b>Decision Tree Classifier:</b></p> <pre>Activity 2.2: Decision Tree Classifier</pre> <pre>In [34]: from sklearn.tree import DecisionTreeClassifier In [35]: dtc = DecisionTreeClassifier() In [36]: dtc.fit(X_train, y_train) In [37]: y_pred = dtc.predict(X_test) In [38]: from sklearn.metrics import accuracy_score, confusion_matrix, classification_report, roc_auc_score, roc_curve In [39]: accuracy_score(y_test, y_pred) Out[39]: 0.8571428571428571 In [40]: confusion_matrix(y_test, y_pred) Out[40]: array([[10, 0],               [ 0, 10]]) In [41]: print(classification_report(y_test, y_pred)) precision    recall  f1 score   support 0.00         0.00         0.00         10 1.00         1.00         1.00         10 avg / total         0.86         0.86         0.86         20</pre> <p><b>XGBoost classifier:</b></p> <pre>xgbmodel</pre> <pre>In [34]: from sklearn import xgb In [35]: xgb = xgb.XGBClassifier() In [36]: xgb.fit(X_train, y_train) In [37]: y_pred = xgb.predict(X_test) In [38]: from sklearn.metrics import accuracy_score, confusion_matrix, classification_report, roc_auc_score, roc_curve In [39]: accuracy_score(y_test, y_pred) Out[39]: 0.8571428571428571 In [40]: confusion_matrix(y_test, y_pred) Out[40]: array([[10, 0],               [ 0, 10]]) In [41]: print(classification_report(y_test, y_pred)) precision    recall  f1 score   support 0.00         0.00         0.00         10 1.00         1.00         1.00         10 avg / total         0.86         0.86         0.86         20</pre>

