

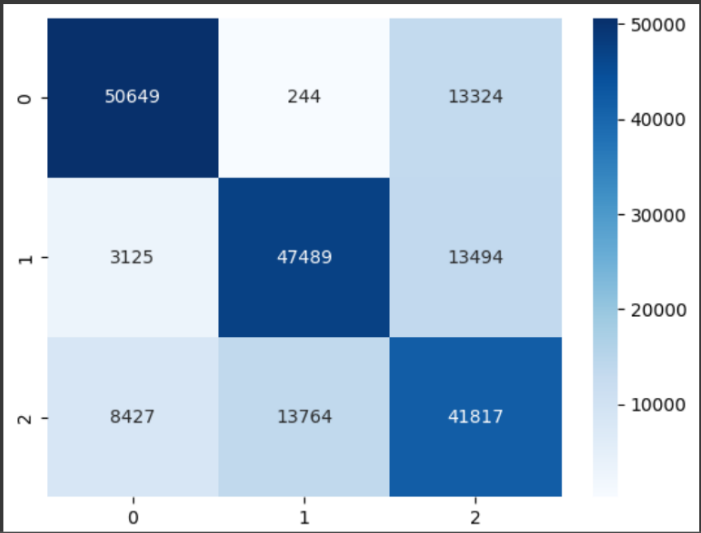
Project Development Phase Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID593014
Project Name	Project- Diabetes Prediction Using Machine Learning
Maximum Marks	10 Marks

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot (Random Forest Model)																																			
1.	Metrics	<p>Regression Model: MAE - , MSE - , RMSE - , R2 score -</p> <p>Classification Model: Confusion Matrix - , Accuracy Score- & Classification Report -</p>	<p>Classification Model:</p> <pre>print("Test accuracy", accuracy_score(y_test,y_pred5)) print("Train accuracy", accuracy_score(y_train_smote,y_pred5_train))</pre> <p>Test accuracy 0.7060469883317565 Train accuracy 0.725540102237643</p> <pre>print(classification_report(y_test,y_pred5))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0.0</td><td>0.93</td><td>0.73</td><td>0.82</td><td>42794</td></tr><tr><td>1.0</td><td>0.03</td><td>0.09</td><td>0.04</td><td>897</td></tr><tr><td>2.0</td><td>0.31</td><td>0.65</td><td>0.42</td><td>7045</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.71</td><td>50736</td></tr><tr><td>macro avg</td><td>0.42</td><td>0.49</td><td>0.43</td><td>50736</td></tr><tr><td>weighted avg</td><td>0.83</td><td>0.71</td><td>0.75</td><td>50736</td></tr></tbody></table>		precision	recall	f1-score	support	0.0	0.93	0.73	0.82	42794	1.0	0.03	0.09	0.04	897	2.0	0.31	0.65	0.42	7045	accuracy			0.71	50736	macro avg	0.42	0.49	0.43	50736	weighted avg	0.83	0.71	0.75	50736
	precision	recall	f1-score	support																																		
0.0	0.93	0.73	0.82	42794																																		
1.0	0.03	0.09	0.04	897																																		
2.0	0.31	0.65	0.42	7045																																		
accuracy			0.71	50736																																		
macro avg	0.42	0.49	0.43	50736																																		
weighted avg	0.83	0.71	0.75	50736																																		

			<pre>from sklearn.metrics import confusion_matrix, classification_report cm5 = confusion_matrix(y_test,y_pred5) sns.heatmap(cm5, annot=True, fmt='d', cmap='Blues', cbar=True) plt.show()</pre>  <table border="1"><thead><tr><th></th><th>0</th><th>1</th><th>2</th></tr></thead><tbody><tr><th>0</th><td>50649</td><td>244</td><td>13324</td></tr><tr><th>1</th><td>3125</td><td>47489</td><td>13494</td></tr><tr><th>2</th><td>8427</td><td>13764</td><td>41817</td></tr></tbody></table>		0	1	2	0	50649	244	13324	1	3125	47489	13494	2	8427	13764	41817
	0	1	2																
0	50649	244	13324																
1	3125	47489	13494																
2	8427	13764	41817																

2.	Tune the Model	Hyperparameter Tuning - Validation Method -	<pre># Define the parameter grid for Random Forest hyperparameters = { "n_estimators": [100, 200, 300], "max_depth": [3, 5, 7], "min_samples_split": [2, 4, 6] } # Create a Random Forest classifier classifier = RandomForestClassifier(random_state=47) # Create a GridSearchCV object and fit it to the data grid_search = GridSearchCV(classifier, hyperparameters, scoring="accuracy", cv=5) grid_search.fit(x_train, y_train) # Print the best parameters and the corresponding accuracy print("Best Parameters: ", grid_search.best_params_) print("Best Accuracy: ", grid_search.best_score_) Best Parameters: {'max_depth': 3, 'min_samples_split': 2, 'n_estimators': 100} Best Accuracy: 0.842148573111432 # Use the best model to make predictions on the test set best_classifier = grid_search.best_estimator_ y_pred = best_classifier.predict(x_test) # Evaluate the model on the test set accuracy = accuracy_score(y_test, y_pred) print("Test Set Accuracy: ", accuracy) Test Set Accuracy: 0.843464206874803</pre> <div><div>GridSearchCV</div><div><div>estimator: RandomForestClassifier</div><div>RandomForestClassifier</div></div></div>
----	----------------	---	--