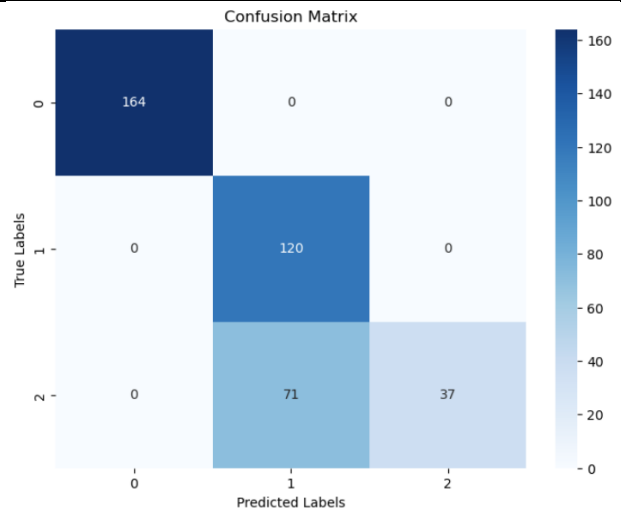


Project Development Phase Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID591582
Project Name	Project - Understanding Audience: A Machine Learning Approach to Customer Segmentation
Maximum Marks	10 Marks

Model Performance Testing:

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

S.No	Parameter	Values	Screenshot																																																																												
1.	Metrics	<div>Classification Model: Confusion Matrix – [[164 0 0] [0 120 0] [0 71 37]] Accuracy Score- Classification Report – <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>1.00</td><td>1.00</td><td>164</td></tr><tr><td>1</td><td>0.63</td><td>1.00</td><td>0.77</td><td>120</td></tr><tr><td>2</td><td>1.00</td><td>0.34</td><td>0.51</td><td>108</td></tr><tr><td>accuracy</td><td>0.82</td><td></td><td></td><td>392</td></tr><tr><td>macro avg</td><td>0.78</td><td>0.76</td><td></td><td>392</td></tr><tr><td>weighted avg</td><td>0.82</td><td>0.80</td><td></td><td>392</td></tr></tbody></table></div>		precision	recall	f1-score	support	0	1.00	1.00	1.00	164	1	0.63	1.00	0.77	120	2	1.00	0.34	0.51	108	accuracy	0.82			392	macro avg	0.78	0.76		392	weighted avg	0.82	0.80		392	<div><div>Confusion Matrix True Labels Predicted Labels</div><div><table><thead><tr><th></th><th>Testing Accuracy</th><th>Validation Accuracy</th></tr></thead><tbody><tr><td>0</td><td>0.787356</td><td>0.818878</td></tr></tbody></table></div><div><table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>1.00</td><td>1.00</td><td>164</td></tr><tr><td>1</td><td>0.63</td><td>1.00</td><td>0.77</td><td>120</td></tr><tr><td>2</td><td>1.00</td><td>0.34</td><td>0.51</td><td>108</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.82</td><td>392</td></tr><tr><td>macro avg</td><td>0.88</td><td>0.78</td><td>0.76</td><td>392</td></tr><tr><td>weighted avg</td><td>0.89</td><td>0.82</td><td>0.80</td><td>392</td></tr></tbody></table></div></div>		Testing Accuracy	Validation Accuracy	0	0.787356	0.818878		precision	recall	f1-score	support	0	1.00	1.00	1.00	164	1	0.63	1.00	0.77	120	2	1.00	0.34	0.51	108	accuracy			0.82	392	macro avg	0.88	0.78	0.76	392	weighted avg	0.89	0.82	0.80	392
	precision	recall	f1-score	support																																																																											
0	1.00	1.00	1.00	164																																																																											
1	0.63	1.00	0.77	120																																																																											
2	1.00	0.34	0.51	108																																																																											
accuracy	0.82			392																																																																											
macro avg	0.78	0.76		392																																																																											
weighted avg	0.82	0.80		392																																																																											
	Testing Accuracy	Validation Accuracy																																																																													
0	0.787356	0.818878																																																																													
	precision	recall	f1-score	support																																																																											
0	1.00	1.00	1.00	164																																																																											
1	0.63	1.00	0.77	120																																																																											
2	1.00	0.34	0.51	108																																																																											
accuracy			0.82	392																																																																											
macro avg	0.88	0.78	0.76	392																																																																											
weighted avg	0.89	0.82	0.80	392																																																																											

2.	Tune the Model	<p>Hyperparameter Tuning -</p> <pre>'n_estimators': [3, 4, 5, 6, 7, 8, 9, 10], 'learning_rate': [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0], 'estimator': [None, DecisionTreeClassifier(max_depth=1)],</pre>	<pre># Define the hyperparameter grid param_grid = { 'n_estimators': [3, 4, 5, 6, 7, 8, 9, 10], 'learning_rate': [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0], 'estimator': [None, DecisionTreeClassifier(max_depth=1)], } # Use GridSearchCV for hyperparameter tuning grid_search = GridSearchCV(adaboost_model, param_grid, cv=5, scoring='accuracy') grid_search.fit(X_train, y_train) best_params = grid_search.best_params_ best_adaboost = AdaBoostClassifier(**best_params) best_adaboost.fit(X_train, y_train) # Predictions y_pred = best_adaboost.predict(X_test) # Model performance accuracy = accuracy_score(y_test, y_pred) print("Best Hyperparameters:", best_params) print("Accuracy on Test Set:", accuracy)</pre>
----	----------------	---	--