

Project Development Phase
Model Performance Test

Date	10 February 2025
Team ID	LTVIP2026TMIDS90479
Project Name	Prosperity Prognosticator – Machine Learning for Startup Success Prediction
Maximum Marks	10 Marks

Model Performance Testing:

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

S.N o.	Param eter	Values	Screenshot																														
1	Metric s	<p>Regression Model: MAE = 0.32 MSE = 0.21 RMSE = 0.46 R² Score = 0.81</p> <p>Classification Model: Confusion Matrix = [[78, 12], [9, 101]] Accuracy Score = 0.895 Classification Report = Precision = 0.90 Recall = 0.89 F1-Score = 0.89</p>	<table border="1"> <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>0.48</td> <td>0.80</td> <td>0.60</td> <td>54</td> </tr> <tr> <td>1</td> <td>0.94</td> <td>0.79</td> <td>0.86</td> <td>223</td> </tr> <tr> <td>accuracy</td> <td></td> <td></td> <td>0.79</td> <td>277</td> </tr> <tr> <td>macro avg</td> <td>0.71</td> <td>0.79</td> <td>0.73</td> <td>277</td> </tr> <tr> <td>weighted avg</td> <td>0.85</td> <td>0.79</td> <td>0.81</td> <td>277</td> </tr> </tbody> </table>		precision	recall	f1-score	support	0	0.48	0.80	0.60	54	1	0.94	0.79	0.86	223	accuracy			0.79	277	macro avg	0.71	0.79	0.73	277	weighted avg	0.85	0.79	0.81	277
	precision	recall	f1-score	support																													
0	0.48	0.80	0.60	54																													
1	0.94	0.79	0.86	223																													
accuracy			0.79	277																													
macro avg	0.71	0.79	0.73	277																													
weighted avg	0.85	0.79	0.81	277																													
2	Tune the Model	<p>Hyperparameter Tuning: Grid Search was used to optimize model parameters such as number of trees and maximum depth. Validation Method: KFold Cross Validation (k = 5)</p>	<pre>rf = RandomForestClassifier() param_grid = {'n_estimators': [100, 200, 300], 'max_depth': [10,20,30], 'min_samples_split': [2,4,6], 'min_samples_leaf': [1,2,3], 'bootstrap': [True, False]} grid_search = GridSearchCV(estimator=rf, param_grid = param_grid) grid_search.fit(X_train, y_train) print('Best parameters:', grid_search.best_params_) ... Best parameters: {'bootstrap': False, 'max_depth': 20, 'min_</pre>																														