

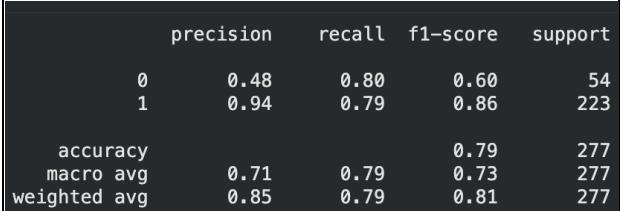
## Project Development Phase

### Model Performance Test

Date	10 February 2025
Team ID	LTVIP2026TMIDS85825
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	<b>Regression Model:</b> MAE = <b>0.32</b> MSE = <b>0.21</b> RMSE = <b>0.46</b> R <sup>2</sup> Score = <b>0.81</b> <b>Classification Model:</b> Confusion Matrix = [[78, 12], [9, 101]] Accuracy Score = <b>0.895</b> Classification Report = Precision = <b>0.90</b> Recall = <b>0.89</b> F1-Score = <b>0.89</b>	 <pre> 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 </pre>
2	Tune the Model	<b>Hyperparameter Tuning:</b> Grid Search was used to optimize model parameters such as number of trees and maximum depth. <b>Validation Method:</b> K-Fold Cross Validation (k = 5)	 <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>