

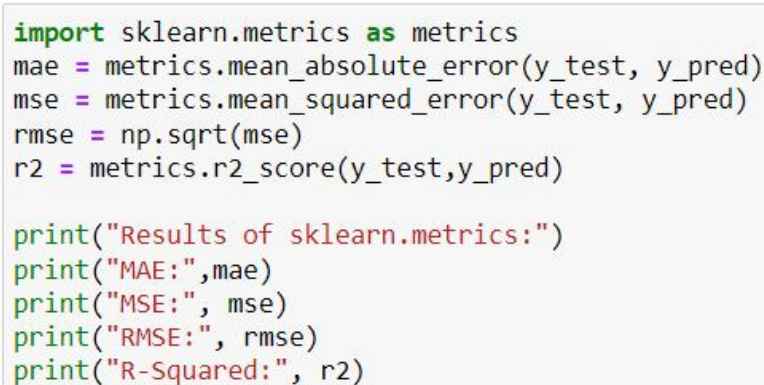
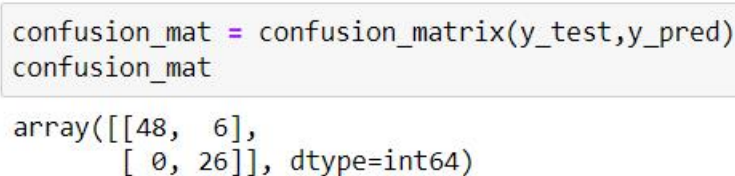
Project Development Phase

Model Performance Test

Date	19 November 2022
Team ID	PNT2022TMID00056
Project Name	Early Detection of Chronic Kidney Disease 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
1.	Metrics	<p>Regression Model: MAE - 0.075 MSE - 0.075 RMSE - 0.27386 R2 score - 0.65811</p> <p>Classification Model: Confusion Matrix</p>	 <pre>import sklearn.metrics as metrics mae = metrics.mean_absolute_error(y_test, y_pred) mse = metrics.mean_squared_error(y_test, y_pred) rmse = np.sqrt(mse) r2 = metrics.r2_score(y_test,y_pred) print("Results of sklearn.metrics:") print("MAE:",mae) print("MSE:", mse) print("RMSE:", rmse) print("R-Squared:", r2)</pre> <p>Results of sklearn.metrics: MAE: 0.075 MSE: 0.075 RMSE: 0.27386127875258304 R-Squared: 0.6581196581196581</p>  <pre>confusion_mat = confusion_matrix(y_test,y_pred) confusion_mat array([[48, 6], [0, 26]], dtype=int64)</pre>

		Accuracy Score	<pre>accuracy_score(y_test,y_pred)</pre> 0.925																														
		Classification Report	<pre>from sklearn.metrics import classification_report print(classification_report(y_test, y_pred))</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</td><td>1.00</td><td>0.89</td><td>0.94</td><td>54</td></tr><tr><td>1</td><td>0.81</td><td>1.00</td><td>0.90</td><td>26</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.93</td><td>80</td></tr><tr><td>macro avg</td><td>0.91</td><td>0.94</td><td>0.92</td><td>80</td></tr><tr><td>weighted avg</td><td>0.94</td><td>0.93</td><td>0.93</td><td>80</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	0.89	0.94	54	1	0.81	1.00	0.90	26	accuracy			0.93	80	macro avg	0.91	0.94	0.92	80	weighted avg	0.94	0.93	0.93	80
	precision	recall	f1-score	support																													
0	1.00	0.89	0.94	54																													
1	0.81	1.00	0.90	26																													
accuracy			0.93	80																													
macro avg	0.91	0.94	0.92	80																													
weighted avg	0.94	0.93	0.93	80																													
2.	Tune the Model	Hyperparameter Tuning - GridSearch CV with best score 0.9175	<pre>from sklearn.model_selection import GridSearchCV c_space = np.logspace(-5, 8, 15) param_grid = {'C': c_space} logreg = LogisticRegression() logreg_cv = GridSearchCV(logreg, param_grid, cv = 5) logreg_cv.fit(x, y) print("Tuned Logistic Regression Parameters: {}".format(logreg_cv.best_params_)) print("Best score is {}".format(logreg_cv.best_score_))</pre> <div>Tuned Logistic Regression Parameters: {'C': 268.2695795279727} Best score is 0.9175000000000001</div>																														