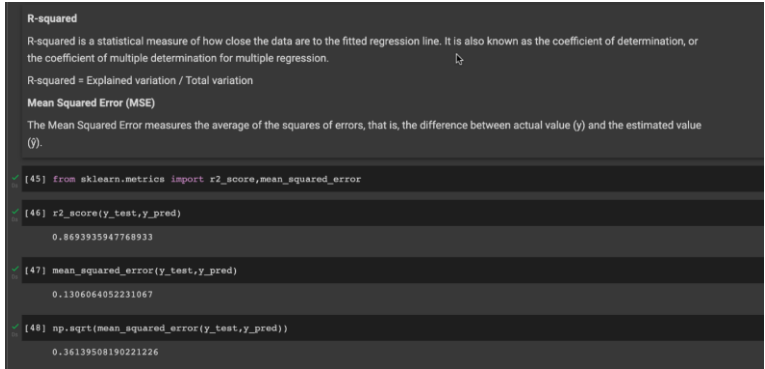


## Project Development Phase Model Performance Test

Date	14 November 2022
Team ID	PNT2022TMID41512
Project Name	Project - Machine Learning based Vehicle Performance Analyser
Maximum Marks	10 Marks

### Model Performance Testing:

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

S.No.	Parameter	Values	Screenshot
1.	Metrics	<p><b>Regression Model:</b> MAE - , MSE - , RMSE - , R2 score -</p> <p><b>Classification Model:</b> Confusion Matrix - , Accuray Score- &amp; Classification Report -</p>	<p>Decision tree regressor</p>  <pre> R-squared R-squared is a statistical measure of how close the data are to the fitted regression line. It is also known as the coefficient of determination, or the coefficient of multiple determination for multiple regression. R-squared = Explained variation / Total variation Mean Squared Error (MSE) The Mean Squared Error measures the average of the squares of errors, that is, the difference between actual value (y) and the estimated value (ŷ).  [45] from sklearn.metrics import r2_score, mean_squared_error [46] r2_score(y_test, y_pred) 0.8693935947768933 [47] mean_squared_error(y_test, y_pred) 0.1306064052231067 [48] np.sqrt(mean_squared_error(y_test, y_pred)) 0.36139508190221226 </pre>

			<p>Random forest regressor</p> <pre> from sklearn.metrics import r2_score, mean_squared_error  [54] r2_score(y_test, y_pred2) 0.9172449209441422  [55] mean_squared_error(y_test, y_pred2) 0.08275507905585772  [56] np.sqrt(mean_squared_error(y_test, y_pred2)) 0.28767182527292745 </pre> <p>Linear regression</p> <pre> [ ] from sklearn.metrics import r2_score, mean_squared_error     r2_score(y_test, y_pred3) 0.8631101197005312  [ ] mean_squared_error(y_test, y_pred3) 0.1368898802994688  np.sqrt(mean_squared_error(y_test, y_pred3)) 0.3699863244762822 </pre> <p>Conclusion:</p> <p>When comparing models, the model with the higher R-squared value is a better fit for the data.</p> <p>When comparing models, the model with the smallest MSE value is a better fit for the data.</p> <p>Comparing these three models, we conclude that the DecisionTree model is the best model to be able to predict mpg from our dataset.</p>
2.	Tune the Model	Hyperparameter Tuning - Validation Method -	<pre> from sklearn.preprocessing import StandardScaler sd = StandardScaler() x_train = sd.fit_transform(x_train) x_test = sd.fit_transform(x_test) y_train = sd.fit_transform(y_train) y_test = sd.fit_transform(y_test)  x_train </pre> <pre> array([[ 1.46858608,  2.48230464,  2.97856512,  1.62455076, -1.61295698,         -0.71873488],        [ 1.46858608,  1.48729292,  1.55429873,  0.84358808, -1.61295698,         -0.71873488],        [-0.86550411, -0.70364636, -0.64684023, -0.36507278,  0.82235108,         -0.71873488],        ...,        [-0.86550411, -1.21071964, -1.44960856, -1.31380657, -0.80118763,          0.53032865],        [ 0.30154098,  0.53055088, -0.12892518,  0.35799706, -1.3423672 ,         -0.71873488],        [-0.86550411, -1.00023639, -0.87990201, -0.89319732, -0.26000806,          0.53032865]]) </pre>

