



Model Optimization and Tuning Phase Template

Date	10 July 2024
Team ID	739659
Project Title	Trip-Based Modelling of Fuel Consumption in Modern Fleet Vehicles Using Machine Learning
Maximum Marks	10 Marks

Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

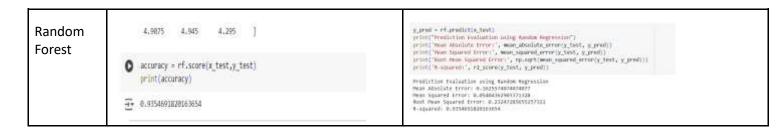
Hyperparameter Tuning Documentation (6 Marks):

Model	Tuned Hyperparameters	Optimal Values
Linear Regression	No Hyperparameters used	
Lasso Regression	No Hyperparameters used	
SVM	No Hyperparameters used	
Decision Tree	No Hyperparameters used	
Random Forest	No Hyperparameters used	

Model	Accuracy	Metrics
Linear Regression	print(HinReg.coef_,linReg.intercept_) [[58] print("Prediction Evaluation using Linear Regression") print("News Absolute Errors", mean_absolute errors(y_test, y_pred)) print("News Absolute Errors", mean_absolute error(y_test, y_pred)) print("News Absolute Errors", mean_absolute errors(y_test, y_pred)) print("Newspaced", rZ_Surer(y_test, y_pred))
Lasso Regression	accuracy = lassoReg.score(x_test,y_test) print(accuracy) 0.1456341532515728	y_pred = lassoReg.predict(x_test) print("brediction Evaluation using lasso Regression") print("brediction Evaluation using lasso Regression") print("beam Absolute Error!", meam absolute error(y_test, y_pred)) print("beam Squared Error"; meam squared error(y_test, y_pred))) print("beam Squared Error"; mp.sqrt(meam aquared error(y_test, y_pred))) print("beam Squared Error"; np.sqrt(meam aquared error(y_test, y_pred))) Prediction Evaluation using lasso Regression Meam Absolute Error: 0.6196444264267669 Meam Squared Error: 0.619634810818613058 Root Ream Squared Error: 0.6183830818613058 R-squared: 0.1456341332515728
SVM	{ (#1} even MCDEL { (#1} ser = SVM().fit(x,y) { (#3}	y_pred = sur.predict(x_test) print("Frediction Evaluation using sur Regression") print("Hean Absolute Error:", mean_absolute error(y_test, y_pred)) print("Hean Squared Error:", mean_aquared_error(y_test, y_pred)) print("Noor Fean Squared Error:", np.sqrt(mean_squared_error(y_test, y_pred))) print("N-squared:", r2_score(y_test, y_pred)) Prediction Evaluation using sur Regression Mean Absolute Error: 0.487119579528160 Mean Squared Error: 0.4871157102488615 Moot Mean squared Error: 0.48884928828772 R-squared: 0.4176454853391483
Decision Tree	[14] 2 - position refrequence (marks, plate + 4)	y prod - st.predict(x text) print("Prodiction bealsation soing deciziontose magnession") print("Prodiction bealsation soing deciziontose magnession") print("New Advantami Street", seas, squared error(y, text, y, prod)) print("New Mouse of Prod") on spiritions, squared error(y, text, y, prod)) print("A nomewell", r2_acpre(y_text, y_prod)) Prodiction bealsation using deciziontose Negrossion Proc Absolute trave s. discommensors Proc Absolute trave s. di







Performance Metrics Comparison Report (2 Marks):

Final Model Selection Justification (2 Marks):

Final Model Selection	Reasoning
Decision Tree	Decision Tree model was selected for its superior performance, exhibiting high accuracy than any other models . We chose the decision tree because it gives very accurate predictions, can handle complex patterns in data, and avoids overfitting. It works well with different types of data and allows us to see which features are most important. This makes it a reliable and effective model for our task