

## Model Development 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	4 Marks

### Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

### Initial Model Training Code:

Paste the screenshot of the model training code

```

✓ [18] from sklearn.model_selection import train_test_split
0s      from sklearn.linear_model import LinearRegression

#splitting Data Into Train And Test

x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=42)
  
```

Linear regression

```


✓ [25] #Linear Regression
0s

▶ linReg = LinearRegression()
  linReg.fit(x_train,y_train)

🔗 ▼ LinearRegression
   LinearRegression()
  
```

## Lasso Regression

✓ [39] #Lasso Regression model


✓ 0s  `lassoReg = linear_model.Lasso(alpha = 0.1)`  
`lassoReg.fit(x,y)`



▼ Lasso  
Lasso(alpha=0.1)

## Decision Tree

✓ [47] #Decision Tree Model:

✓ 0s  `dt = DecisionTreeRegressor(random_state = 0)`  
`dt.fit(x,y)`



▼ DecisionTreeRegressor  
DecisionTreeRegressor(random\_state=0)

## Random Forest









✓ [51] #Random Forest Model:

✓ 0s `rf = RandomForestRegressor(n_estimators = 100 , random_state = 0)`  
`rf.fit(x,y)`



▼ RandomForestRegressor  
RandomForestRegressor(random\_state=0)

## Model Validation and Evaluation Report:

Model	Training Report	Accuracy	Metrix
Linear Regression		0.11	
Lasso Regression		0.14	
SVM		0.41	
Decision Tree		0.98	

Random forest

[illegible]

0.93

```
y_pred = rf.predict(X_test)
print("Model evaluation using Random Regression")
print("Mean Absolute Error: ", mean_absolute_error(y_test, y_pred))
print("Mean Squared Error: ", mean_squared_error(y_test, y_pred))
print("Root Mean Squared Error: ", np.sqrt(mean_squared_error(y_test, y_pred)))
print("R-squared: ", r2_score(y_test, y_pred))

#Model evaluation using Random Regression
Mean Absolute Error: 0.16257307404077
Mean Squared Error: 0.0404747960311518
Root Mean Squared Error: 0.2011828000057321
R-squared: 0.935401028053054
```