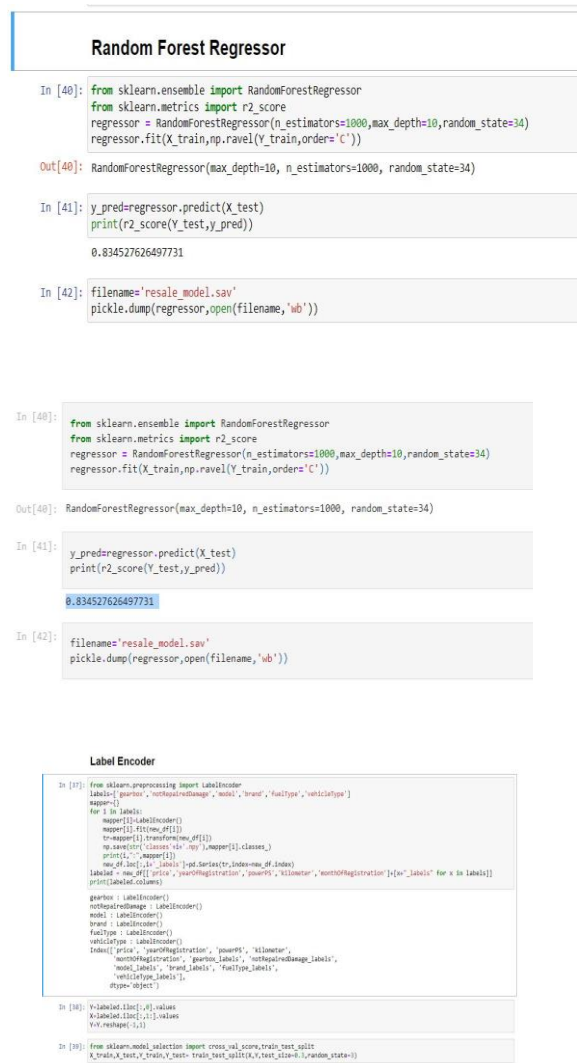


Project Development Phase Model Performance Test

Date	19 November 2022
Team ID	PNT2022TMID02356
Project Name	Project - car resale value prediction
Maximum Marks	10 Marks

Model Performance Testing:

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

S.N o.	Parameter	Values	Screenshot
1.	Metrics	<p>Regression Model: MAE - , MSE - , RMSE - , R2 score -</p> <p>Classification Model: Confusion Matrix - , Accuray Score- & Classification Report -</p> <p>Label Encoder</p>	 <p>The screenshot displays a Jupyter Notebook with the following content:</p> <p>Random Forest Regressor</p> <pre>In [40]: from sklearn.ensemble import RandomForestRegressor from sklearn.metrics import r2_score regressor = RandomForestRegressor(n_estimators=1000,max_depth=10,random_state=34) regressor.fit(X_train,np.ravel(Y_train,order='C')) Out[40]: RandomForestRegressor(max_depth=10, n_estimators=1000, random_state=34) In [41]: y_pred=regressor.predict(X_test) print(r2_score(Y_test,y_pred)) 0.834527626497731 In [42]: filename='resale_model.sav' pickle.dump(regressor,open(filename,'wb'))</pre> <p>Label Encoder</p> <pre>In [37]: from sklearn.preprocessing import LabelEncoder Labels=['gearbox', 'notfourredrange', 'model', 'brand', 'fueltype', 'vehicletype'] mapper={} for i in Labels: mapper[i]=LabelEncoder() mapper[i].fit(Y_train[df[i]]) tr_mapper[i]=transformer[df[i]] m_mapper[i]=mapper[i].classes_ print(i," ",mapper[i]) new_df=df[['price','yearOfRegistration','year99','kilometer','monthOfRegistration']+['%s' % i for i in Labels]] labeled = new_df[['price','yearOfRegistration','year99','kilometer','monthOfRegistration']+['%s' % i for i in Labels]] print(labeled.columns) gearbox = LabelEncoder() notfourredrange = LabelEncoder() model = LabelEncoder() brand = LabelEncoder() fueltype = LabelEncoder() vehicletype = LabelEncoder() Index(['price', 'yearOfRegistration', 'year99', 'kilometer', 'monthOfRegistration', 'gearbox_labels', 'notfourredrange_labels', 'model_labels', 'brand_labels', 'fueltype_labels', 'vehicletype_labels'], dtype=object) In [38]: Y=labeled.iloc[:,0].values X=labeled.iloc[:,1:].values X=X.reshape(-1,1) In [39]: from sklearn.model_selection import cross_val_score,train_test_split X_train,X_test,X_val,X_test=train_test_split(X,X_test,X_val,X_test,random_state=0)</pre>

