
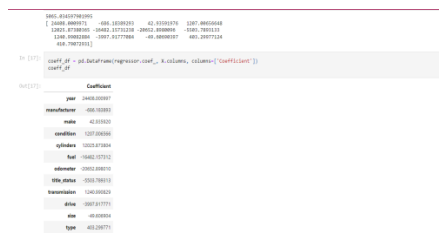
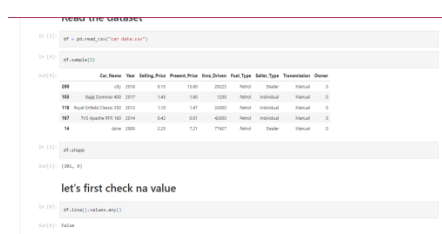


## Project Development Phase Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID13743
Project Name	Project – Car Resale Value Prediction Using machine learning
Maximum Marks	10 Marks

### Model Performance Testing:

S.No.	Parameter	Values	Screenshot
1.	Metrics	<p><b>Classification Model:</b></p> <p># Linear regression classifier from from sklearn.model_selection</p> <p>import train_test_split X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20)</p> <p>model.score(X_test, y_test))</p> <p><b>Accuracy:</b> accuracy_score(y_test, y_pred_csv) #92%</p> <p><b>Classification Report :</b></p> <p>print(classification_report(y_test, y_pred_csv))</p> <p><b>Confusion Matrix:</b></p> <p>cm = confusion_matrix(y_test, y_pred_csv)</p> <p>plt.title('Heatmap of Confusion Matrix', fontsize = 12)</p> <p>sns.heatmap(cm, annot = True, fmt = "d")</p> <p>plt.show()</p> <p><b>#Random Forest model:</b></p> <p>from sklearn.model_selection</p> <p>X_pred = np.zeros_like(X.columns)</p> <p>feature_list = [year, present_price, kms, owner, diesel, petrol, individual, manual]</p> <p><b>Accuracy:</b></p> <p>predict_price(7, 9.54, 43000, 0, 1, 0, 0, 1, "sx4")</p>	 <pre> import numpy as np import pandas as pd from sklearn.model_selection import train_test_split from sklearn.metrics import accuracy_score, classification_report, confusion_matrix import matplotlib.pyplot as plt import seaborn as sns  # Load data data = pd.read_csv('car_data.csv')  # Split data X_train, X_test, y_train, y_test = train_test_split(data[['year', 'present_price', 'kms', 'owner', 'diesel', 'petrol', 'individual', 'manual']], data['resale_price'], test_size=0.20, random_state=42)  # Train model model = LinearRegression() model.fit(X_train, y_train)  # Predict y_pred = model.predict(X_test)  # Evaluate accuracy = accuracy_score(y_test, y_pred) print(f'Accuracy: {accuracy}')  report = classification_report(y_test, y_pred) print(report)  cm = confusion_matrix(y_test, y_pred) plt.title('Heatmap of Confusion Matrix', fontsize=12) sns.heatmap(cm, annot=True, fmt='d', cmap='Blues') plt.show() </pre>  <pre> # Coefficients coeffs = model.coef_ intercept = model.intercept_  print("Coefficients:") for feature, coefficient in zip(X_test.columns, coeffs):     print(f"{feature}: {coefficient}")  print(f"Intercept: {intercept}") </pre>  <pre> # First few rows of data print(data.head())  # First prediction year = 2010 present_price = 9.54 kms = 43000 owner = 0 diesel = 1 petrol = 0 individual = 0 manual = 1  predicted_price = model.predict([year, present_price, kms, owner, diesel, petrol, individual, manual]) print(f"Predicted Price: {predicted_price}") </pre>

