Car Price Prediction Model

Project Title :

Building a Car Price Prediction Model – A Machine Learning Approach

Objective :

The goal is to develop a machine learning model that can accurately estimate the selling price of used cars. You will handle data preprocessing, perform exploratory analysis, train a predictive model, and evaluate its accuracy using statistical metrics.

Dataset :

A dataset containing information about used cars. It includes:

No. of Rows : 4340

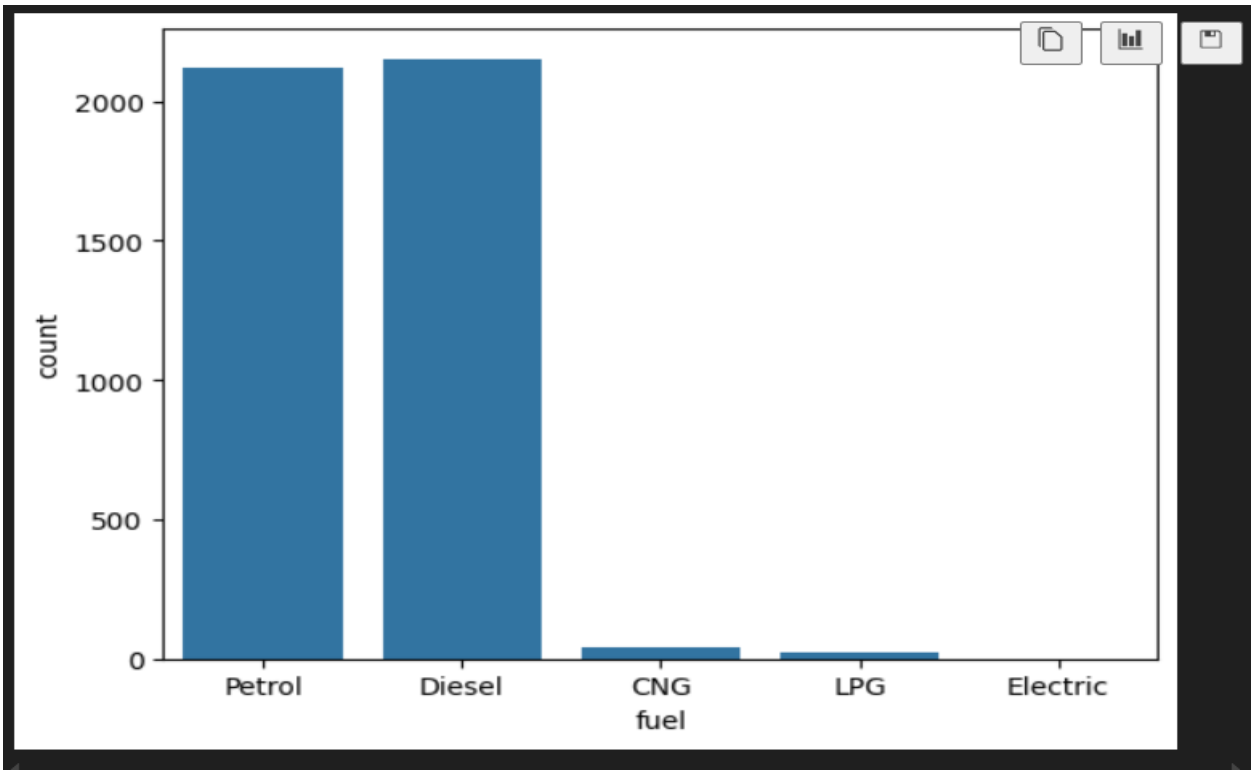
No. of Columns : 8

Column Names : Name, Year, SellingPrice, kmDriven, Fuel, SellerType, Transmission, Owner.

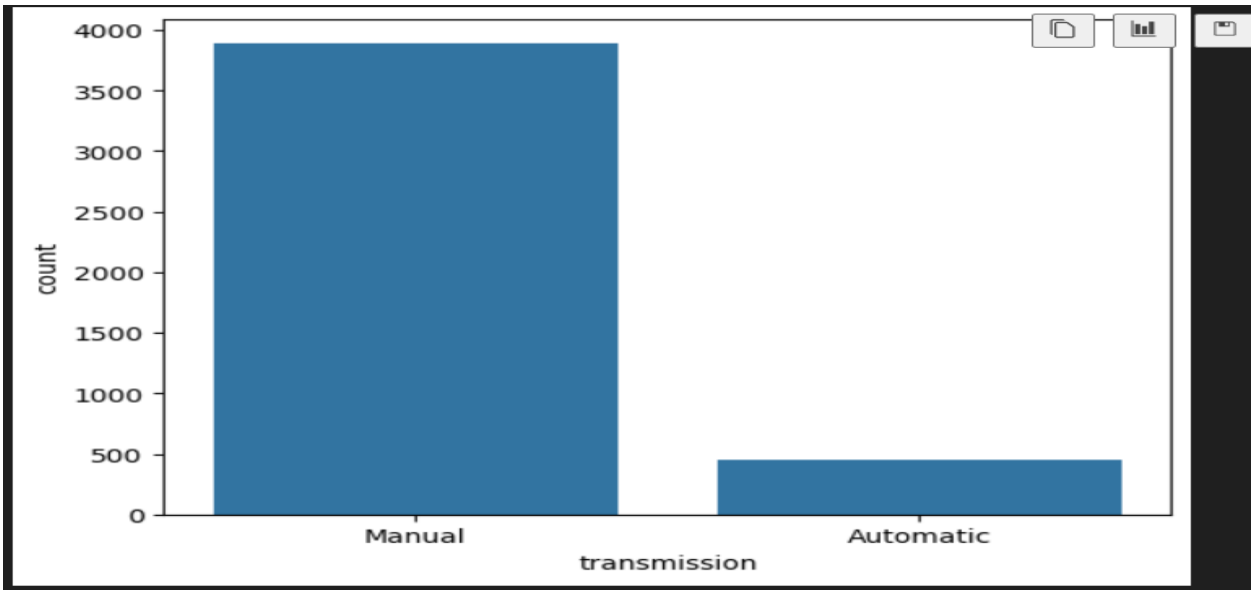
Categorical Columns : Name, Fuel, SellerType, Transmission, Owner

Numerical Columns : Year, Selling Price, kmDriven

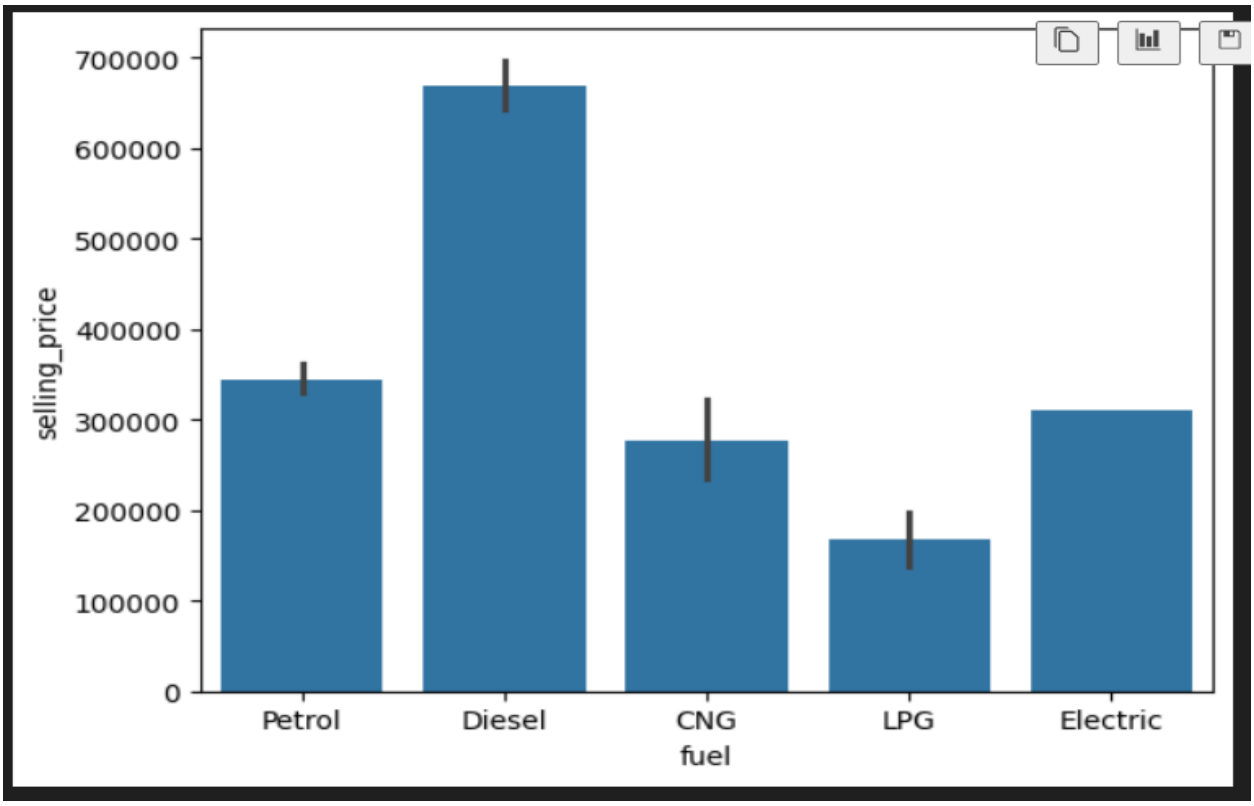
Analysis of Dataset :



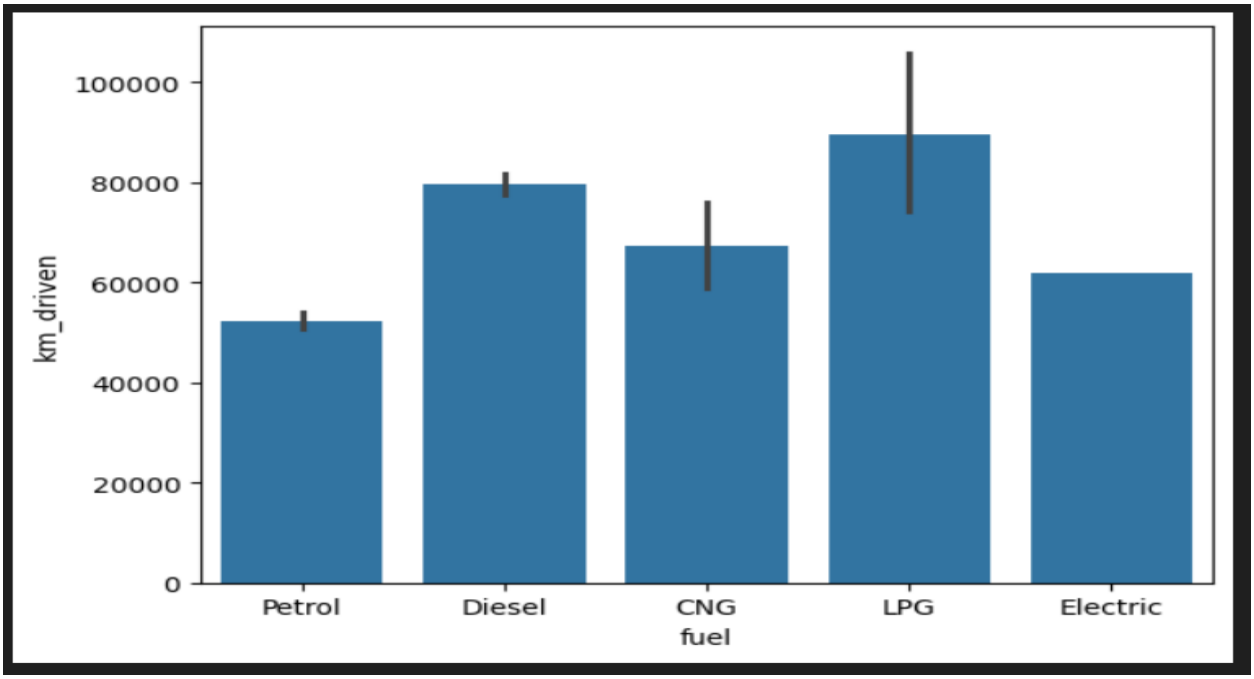
This graph tells us that we have demand of Petrol and diesel Cars in comparison to Cars of CNG,LPG or Electric.



This graph tells us that most of the peoples purchase manual cars instead of automatic.



This graph tells us that what is the average price on which a car we can sell.



As we can see in this, LPG Cars are more driven in comparison to Diesel Cars.

Model Preparation

```
X = cars.drop(columns=['selling_price'])
y = cars['selling_price']
```
✓ 0.0s

Here we Differentiate DataSet into two part first one is X in which we give our all feature columns and in second one we give our target column which is Selling Price.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3,random_state=42)
```
✓ 0.0s

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score,mean_absolute_error,mean_squared_error,root_mean_squared_error
lr = LinearRegression()
```
✓ 0.0s

```
lr.fit(X_train,y_train)
```
✓ 0.0s

LinearRegression
LinearRegression()

Here we divide our data into train and test part.

We use 70% data to train our model and 30% data to test our model.

Here we use Linear Regression model to predict our output column.

After All training our linear regression model…..

Model Performance is this :

```
y_pred = lr.predict(X_test)
```
✓ 0.0s

```
print('MAE : ',mean_absolute_error(y_test,y_pred))
```
✓ 0.0s

```
MAE :  220998.98142815338
```

```
print('MSE : ',mean_squared_error(y_test,y_pred))
```
✓ 0.0s

```
MSE :  164823320437.4607
```

```
print('RMSE : ',root_mean_squared_error(y_test,y_pred))
```
✓ 0.0s

```
RMSE :  405984.38447489665
```

```
print('R2 Score : ',r2_score(y_test,y_pred))
```
✓ 0.0s

```
R2 Score :  0.4425257309037879
```

So our models Errors :

Mean Absolute Error : 220998.9814

Mean Squared Error : 164823320437.4607

Mean Absolute Error : 405984.3844

R2 Score : 0.4425

After seeing these results I tried some other method and these are the result of them :

Model - 1 :
R2 Score = 0.4425257309037879

Model - 2 :
if we select random_state=42
R2 Score = 0.48794141038252026

if we select random_state=50
R2 Score = 0.6168901213295587

Model - 3 :

i) Linear Regression Model
R2 Score = 0.44829675104686706

ii) Random Forest Regression Model
R2 Score = 0.44829675104686706