

```
In [13]: import pandas as pd
import warnings
warnings.filterwarnings("ignore")
```

```
In [14]: data=pd.read_csv("/home/placement/Downloads/fiat500.csv")
```

```
In [24]: data.head(10)
```

Out[24]:

| | ID | model | engine_power | age_in_days | km | previous_owners | lat | lon | price |
|---|----|--------|--------------|-------------|--------|-----------------|-----------|-----------|-------|
| 0 | 1 | lounge | 51 | 882 | 25000 | 1 | 44.907242 | 8.611560 | 8900 |
| 1 | 2 | pop | 51 | 1186 | 32500 | 1 | 45.666359 | 12.241890 | 8800 |
| 2 | 3 | sport | 74 | 4658 | 142228 | 1 | 45.503300 | 11.417840 | 4200 |
| 3 | 4 | lounge | 51 | 2739 | 160000 | 1 | 40.633171 | 17.634609 | 6000 |
| 4 | 5 | pop | 73 | 3074 | 106880 | 1 | 41.903221 | 12.495650 | 5700 |
| 5 | 6 | pop | 74 | 3623 | 70225 | 1 | 45.000702 | 7.682270 | 7900 |
| 6 | 7 | lounge | 51 | 731 | 11600 | 1 | 44.907242 | 8.611560 | 10750 |
| 7 | 8 | lounge | 51 | 1521 | 49076 | 1 | 41.903221 | 12.495650 | 9190 |
| 8 | 9 | sport | 73 | 4049 | 76000 | 1 | 45.548000 | 11.549470 | 5600 |
| 9 | 10 | sport | 51 | 3653 | 89000 | 1 | 45.438301 | 10.991700 | 6000 |

```
In [16]: data1=data.loc[(data.previous_owners==1)]
```

```
In [17]: data1
```

```
Out[17]:
```

| | ID | model | engine_power | age_in_days | km | previous_owners | lat | lon | price |
|------|------|--------|--------------|-------------|--------|-----------------|-----------|-----------|-------|
| 0 | 1 | lounge | 51 | 882 | 25000 | 1 | 44.907242 | 8.611560 | 8900 |
| 1 | 2 | pop | 51 | 1186 | 32500 | 1 | 45.666359 | 12.241890 | 8800 |
| 2 | 3 | sport | 74 | 4658 | 142228 | 1 | 45.503300 | 11.417840 | 4200 |
| 3 | 4 | lounge | 51 | 2739 | 160000 | 1 | 40.633171 | 17.634609 | 6000 |
| 4 | 5 | pop | 73 | 3074 | 106880 | 1 | 41.903221 | 12.495650 | 5700 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 1533 | 1534 | sport | 51 | 3712 | 115280 | 1 | 45.069679 | 7.704920 | 5200 |
| 1534 | 1535 | lounge | 74 | 3835 | 112000 | 1 | 45.845692 | 8.666870 | 4600 |
| 1535 | 1536 | pop | 51 | 2223 | 60457 | 1 | 45.481541 | 9.413480 | 7500 |
| 1536 | 1537 | lounge | 51 | 2557 | 80750 | 1 | 45.000702 | 7.682270 | 5990 |
| 1537 | 1538 | pop | 51 | 1766 | 54276 | 1 | 40.323410 | 17.568270 | 7900 |

1389 rows × 9 columns

```
In [18]: data1=data.drop(['lat','lon','lon'],axis=1)
```

```
In [19]: data1=pd.get_dummies(data1)
```

In [20]: data1

Out[20]:

| | ID | engine_power | age_in_days | km | previous_owners | price | model_lounge | model_pop | model_sport |
|------|------|--------------|-------------|--------|-----------------|-------|--------------|-----------|-------------|
| 0 | 1 | 51 | 882 | 25000 | 1 | 8900 | 1 | 0 | 0 |
| 1 | 2 | 51 | 1186 | 32500 | 1 | 8800 | 0 | 1 | 0 |
| 2 | 3 | 74 | 4658 | 142228 | 1 | 4200 | 0 | 0 | 1 |
| 3 | 4 | 51 | 2739 | 160000 | 1 | 6000 | 1 | 0 | 0 |
| 4 | 5 | 73 | 3074 | 106880 | 1 | 5700 | 0 | 1 | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 1533 | 1534 | 51 | 3712 | 115280 | 1 | 5200 | 0 | 0 | 1 |
| 1534 | 1535 | 74 | 3835 | 112000 | 1 | 4600 | 1 | 0 | 0 |
| 1535 | 1536 | 51 | 2223 | 60457 | 1 | 7500 | 0 | 1 | 0 |
| 1536 | 1537 | 51 | 2557 | 80750 | 1 | 5990 | 1 | 0 | 0 |
| 1537 | 1538 | 51 | 1766 | 54276 | 1 | 7900 | 0 | 1 | 0 |

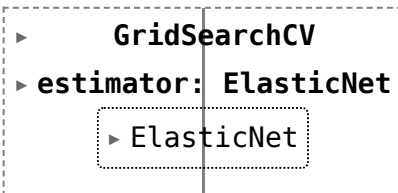
1538 rows × 9 columns

In [21]: `y=data1['price']`
`x=data1.drop('price',axis=1)`

In [22]: `from sklearn.model_selection import train_test_split`
`X_train,X_test,Y_train,Y_test=train_test_split(x,y,test_size=0.33,random_state=42)`

```
In [26]: from sklearn.model_selection import GridSearchCV
        from sklearn.linear_model import ElasticNet
        elastic = ElasticNet()
        parameters={'alpha':[1e-15,1e-10,1e-8,1e-4,1e-3,1e-2,1,5,10,20,30]}
        elastic_regressor=GridSearchCV(elastic,parameters)
        elastic_regressor.fit(X_train,Y_train)
```

Out[26]:



```

  ▸ GridSearchCV
  ▸ estimator: ElasticNet
    ▸ ElasticNet
```

```
In [27]: elastic_regressor.best_params_
```

Out[27]: {'alpha': 0.01}

```
In [28]: elastic=ElasticNet(alpha=0.1)
        elastic.fit(X_train,Y_train)
        y_pred_elastic=elastic.predict(X_test)
```

```
In [29]: from sklearn.metrics import r2_score
        r2_score(Y_test,y_pred_elastic)
```

Out[29]: 0.8414565299012147

```
In [30]: from sklearn.metrics import mean_squared_error
elastic_Error=mean_squared_error(y_pred_elastic,Y_test)
elastic_Error
```

Out[30]: 582240.9011940917

```
In [31]: Results=pd.DataFrame(columns=['price','predicted'])
Results['price']=Y_test
Results['predicted']=y_pred_elastic
Results=Results.reset_index()
Results['ID']=Results.index
Results
```

Out[31]:

| | index | price | predicted | ID |
|-----|-------|-------|--------------|-----|
| 0 | 481 | 7900 | 5903.123297 | 0 |
| 1 | 76 | 7900 | 7223.403259 | 1 |
| 2 | 1502 | 9400 | 9794.808007 | 2 |
| 3 | 669 | 8500 | 9724.783858 | 3 |
| 4 | 1409 | 9700 | 9974.200594 | 4 |
| ... | ... | ... | ... | ... |
| 503 | 291 | 10900 | 10069.090270 | 503 |
| 504 | 596 | 5699 | 6320.490299 | 504 |
| 505 | 1489 | 9500 | 9959.608673 | 505 |
| 506 | 1436 | 6990 | 8310.780259 | 506 |
| 507 | 575 | 10900 | 10384.079146 | 507 |

508 rows × 4 columns

```
In [33]: import seaborn as sns
import matplotlib.pyplot as plt
sns.lineplot(x='ID', y='price', data=Results.head(50))
sns.lineplot(x='ID', y='predicted', data=Results.head(50))
plt.plot()
```

Out[33]: []

