quikr_car is the raw data

In [175]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [176]:
```
car=pd.read_csv('quikr_car.csv')
car.head()
```

Out[176]:

	name	company	year	Price	kms_driven	fuel_type
0	Hyundai Santro Xing XO eRLX Euro III	Hyundai	2007	80,000	45,000 kms	Petrol
1	Mahindra Jeep CL550 MDI	Mahindra	2006	4,25,000	40 kms	Diesel
2	Maruti Suzuki Alto 800 Vxi	Maruti	2018	Ask For Price	22,000 kms	Petrol
3	Hyundai Grand i10 Magna 1.2 Kappa VTVT	Hyundai	2014	3,25,000	28,000 kms	Petrol
4	Ford EcoSport Titanium 1.5L TDCi	Ford	2014	5,75,000	36,000 kms	Diesel

In [177]:
```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 892 entries, 0 to 891
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   name        892 non-null    object
 1   company     892 non-null    object
 2   year        892 non-null    object
 3   Price       892 non-null    object
 4   kms_driven  840 non-null    object
 5   fuel_type   837 non-null    object
dtypes: object(6)
memory usage: 41.9+ KB
```

In [178]:
```
## lets clean the data because data is not cleaned
## 1.year obj to integer plus waste things are there
## 2.price object to integer comma
```

In [179]:
```
car['Price'].unique()
```

Out[179]:
```
array(['80,000', '4,25,000', 'Ask For Price', '3,25,000', '5,75,000',
       '1,75,000', '1,90,000', '8,30,000', '2,50,000', '1,82,000',
       '3,15,000', '4,15,000', '3,20,000', '10,00,000', '5,00,000',
       '3,50,000', '1,60,000', '3,10,000', '75,000', '1,00,000',
       '2,90,000', '95,000', '1,80,000', '3,85,000', '1,05,000',
       '6,50,000', '6,89,999', '4,48,000', '5,49,000', '5,01,000',
       '4,89,999', '2,80,000', '3,49,999', '2,84,999', '3,45,000',
       '4,99,999', '2,35,000', '2,49,999', '14,75,000', '3,95,000',
       '2,20,000', '1,70,000', '85,000', '2,00,000', '5,70,000',
       '1,10,000', '4,48,999', '18,91,111', '1,59,500', '3,44,999',
       '4,49,999', '8,65,000', '6,99,000', '3,75,000', '2,24,999',
       '12,00,000', '1,95,000', '3,51,000', '2,40,000', '90,000',
       '1,55,000', '6,00,000', '1,89,500', '2,10,000', '3,90,000',
       '1,35,000', '16,00,000', '7,01,000', '2,65,000', '5,25,000',
       '3,72,000', '6,35,000', '5,50,000', '4,85,000', '3,29,500',
       '2,51,111', '5,69,999', '69,999', '2,99,999', '3,99,999',
       '4,50,000', '2,70,000', '1,58,400', '1,79,000', '1,25,000',
       '2,99,000', '1,50,000', '2,75,000', '2,85,000', '3,40,000',
       '70,000', '2,89,999', '8,49,999', '7,49,999', '2,74,999',
```

In [180]:
```
backup=car.copy()
```

In [181]:
```
car.head()
```

Out[181]:

	name	company	year	Price	kms_driven	fuel_type
0	Hyundai Santro Xing XO eRLX Euro III	Hyundai	2007	80,000	45,000 kms	Petrol
1	Mahindra Jeep CL550 MDI	Mahindra	2006	4,25,000	40 kms	Diesel
2	Maruti Suzuki Alto 800 Vxi	Maruti	2018	Ask For Price	22,000 kms	Petrol
3	Hyundai Grand i10 Magna 1.2 Kappa VTVT	Hyundai	2014	3,25,000	28,000 kms	Petrol
4	Ford EcoSport Titanium 1.5L TDCi	Ford	2014	5,75,000	36,000 kms	Diesel

In [182]:
```
##car['year'].str.isnumeric() ## it will remove non numeric year format row
```

In [183]:
```
##car['year']=car['year'].astype(int)
```

In [184]:
```
##car=car[car['Price']!="Ask For Price"]
```

In [185]:
```
##car['Price']=car['Price'].str.replace(',','').astype(int)
```

In [186]:
```
car.shape
```

Out[186]:
```
(892, 6)
```

In [187]:
```
car=car[car['year'].str.isnumeric()]
```

In [188]:
```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 842 entries, 0 to 891
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   name        842 non-null    object
 1   company     842 non-null    object
 2   year        842 non-null    object
 3   Price       842 non-null    object
 4   kms_driven  840 non-null    object
 5   fuel_type   837 non-null    object
dtypes: object(6)
memory usage: 46.0+ KB
```

In [189]:
```
car['year']=car['year'].astype(int)
```

```
car.info()
```

In [ ]:
```

```

In [190]:
```
car=car[car['Price']!="Ask For Price"]
```

In [ ]:
```

```

In [191]:
```
car=car[car['Price']!="Ask For Price"]
```

In [192]:
```
car['Price']=car['Price'].str.replace(',','').astype(int)
```

In [193]:
```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 819 entries, 0 to 891
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   name        819 non-null    object
 1   company     819 non-null    object
 2   year        819 non-null    int32 
 3   Price       819 non-null    int32 
 4   kms_driven  819 non-null    object
 5   fuel_type   816 non-null    object
dtypes: int32(2), object(4)
memory usage: 38.4+ KB
```

In [194]:
```
car['kms_driven']=car['kms_driven'].str.split(' ').str.get(0).str.replace('
```

In [195]:
```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 819 entries, 0 to 891
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   name        819 non-null    object
 1   company     819 non-null    object
 2   year        819 non-null    int32 
 3   Price       819 non-null    int32 
 4   kms_driven  819 non-null    object
 5   fuel_type   816 non-null    object
dtypes: int32(2), object(4)
memory usage: 38.4+ KB
```

In [196]:
```
car=car[car['kms_driven'].str.isnumeric()]
```

In [197]:
```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 817 entries, 0 to 889
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   name        817 non-null    object
 1   company     817 non-null    object
 2   year        817 non-null    int32 
 3   Price       817 non-null    int32 
 4   kms_driven  817 non-null    object
 5   fuel_type   816 non-null    object
dtypes: int32(2), object(4)
memory usage: 38.3+ KB
```

In [198]:
```
car['kms_driven']=car['kms_driven'].astype(int)
```

In [199]:
```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 817 entries, 0 to 889
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   name        817 non-null    object
 1   company     817 non-null    object
 2   year        817 non-null    int32 
 3   Price       817 non-null    int32 
 4   kms_driven  817 non-null    int32 
 5   fuel_type   816 non-null    object
dtypes: int32(3), object(3)
memory usage: 35.1+ KB
```

In [200]:
```
car=car[~car['fuel_type'].isna()]
```

In [201]:
```
car['name']=car['name'].str.split(' ').str.slice(0,3).str.join(' ')
```

In [202]:
```
car['name']
```

Out[202]:
```
0         Hyundai Santro Xing
1         Mahindra Jeep CL550
3           Hyundai Grand i10
4      Ford EcoSport Titanium
6                   Ford Figo
                ...
883        Maruti Suzuki Ritz
885            Tata Indica V2
886      Toyota Corolla Altis
888              Tata Zest XM
889        Mahindra Quanto C8
Name: name, Length: 816, dtype: object
```

In [203]:
```
car.reset_index(drop=True)
```

Out[203]:

	name	company	year	Price	kms_driven	fuel_type
0	Hyundai Santro Xing	Hyundai	2007	80000	45000	Petrol
1	Mahindra Jeep CL550	Mahindra	2006	425000	40	Diesel
2	Hyundai Grand i10	Hyundai	2014	325000	28000	Petrol
3	Ford EcoSport Titanium	Ford	2014	575000	36000	Diesel
4	Ford Figo	Ford	2012	175000	41000	Diesel
...	...	...	...	...	...	...
811	Maruti Suzuki Ritz	Maruti	2011	270000	50000	Petrol
812	Tata Indica V2	Tata	2009	110000	30000	Diesel
813	Toyota Corolla Altis	Toyota	2009	300000	132000	Petrol
814	Tata Zest XM	Tata	2018	260000	27000	Diesel
815	Mahindra Quanto C8	Mahindra	2013	390000	40000	Diesel

816 rows × 6 columns

In [204]:
```
car.describe()#detect outliers in the price
```

Out[204]:

	year	Price	kms_driven
count	816.000000	8.160000e+02	816.000000
mean	2012.444853	4.117176e+05	46275.531863
std	4.002992	4.751844e+05	34297.428044
min	1995.000000	3.000000e+04	0.000000
25%	2010.000000	1.750000e+05	27000.000000
50%	2013.000000	2.999990e+05	41000.000000
75%	2015.000000	4.912500e+05	56818.500000
max	2019.000000	8.500003e+06	400000.000000

In [221]:
```
car=car[car['Price']<6e06].reset_index(drop=True)
```

In [222]:
```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 815 entries, 0 to 814
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   name        815 non-null    object
 1   company     815 non-null    object
 2   year        815 non-null    int32 
 3   Price       815 non-null    int32 
 4   kms_driven  815 non-null    int32 
 5   fuel_type   815 non-null    object
dtypes: int32(3), object(3)
memory usage: 28.8+ KB
```

In [223]:
```
car.to_csv('cleaned_car.csv')## cleaning complete
```

In [224]:
```
x=car.drop(columns='Price')
```

In [225]:
```
y=car['Price']
```

In [226]:
```
x
```

Out[226]:

	name	company	year	kms_driven	fuel_type
0	Hyundai Santro Xing	Hyundai	2007	45000	Petrol
1	Mahindra Jeep CL550	Mahindra	2006	40	Diesel
2	Hyundai Grand i10	Hyundai	2014	28000	Petrol
3	Ford EcoSport Titanium	Ford	2014	36000	Diesel
4	Ford Figo	Ford	2012	41000	Diesel
...	...	...	...	...	...
810	Maruti Suzuki Ritz	Maruti	2011	50000	Petrol
811	Tata Indica V2	Tata	2009	30000	Diesel
812	Toyota Corolla Altis	Toyota	2009	132000	Petrol
813	Tata Zest XM	Tata	2018	27000	Diesel
814	Mahindra Quanto C8	Mahindra	2013	40000	Diesel

815 rows × 5 columns

In [274]:
```
from sklearn.model_selection import train_test_split
```

In [275]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2)
```

In [297]:
```
from sklearn.linear_model import LinearRegression
```

In [298]:
```
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import make_column_transformer
from sklearn.pipeline import make_pipeline
from sklearn.metrics import r2_score
```

In [301]:
```
ohe=OneHotEncoder()
ohe.fit(x[['name','company','fuel_type']])
```

Out[301]:

OneHotEncoder (https://scikit-learn.org/1.4/modules/generated/sklearn.preprocessing.OneHotEncod

OneHotEncoder()

In [302]:
```
column_trans=make_column_transformer((OneHotEncoder(categories=ohe.categori
                                     remainder='passthrough')
```

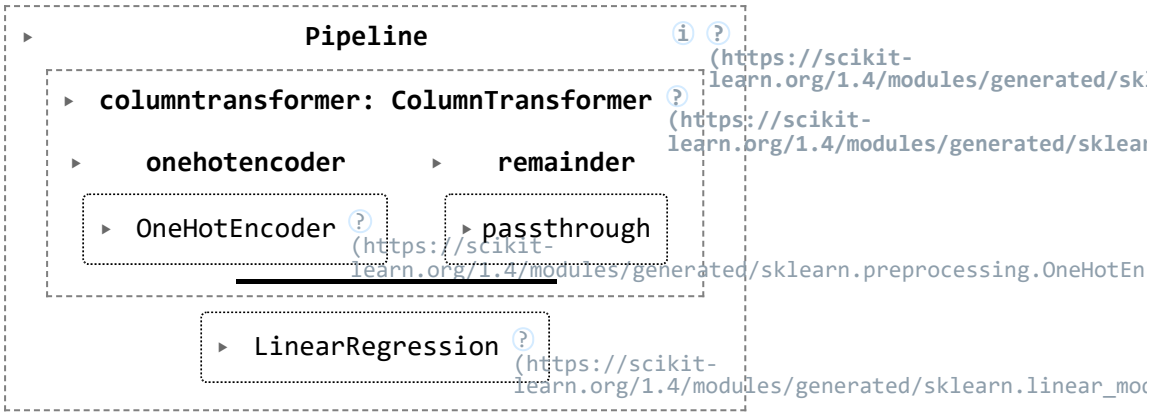
In [288]:
```

```

In [303]:
```
lr=LinearRegression()
```

In [304]:
```
pipe=make_pipeline(column_trans,lr)
```

In [306]:
```
pipe.fit(x_train,y_train)
```

Out[306]:



In [307]:
```
y_pred=pipe.predict(x_test)
```

In [316]:
```
r2_score(y_test,y_pred)
```

Out[316]:
```
0.7435608874206472
```

In [317]:
```
scores=[]
for i in range(1000):
    x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.1,random
    lr=LinearRegression()
    pipe=make_pipeline(column_trans,lr)
    pipe.fit(x_train,y_train)
    y_pred=pipe.predict(x_test)
    scores.append(r2_score(y_test,y_pred))
```

In [320]:
```
np.argmax(scores)
```

Out[320]:
```
302
```

In [322]:
```
max_arg_score=scores[np.argmax(scores)]
max_arg_score
```

Out[322]:
```
0.8991138463319752
```

In [323]:
```
import pickle
```

In [324]:
```
pickle.dump(pipe,open('linearreg_carprice.pkl','wb'))
```

In [326]:
```
pipe.predict(pd.DataFrame(columns=x_test.columns,data=np.array(['Maruti Suz
```

Out[326]:
```
array([579459.7897751])
```

In [ ]: