

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn import preprocessing, svm
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

```
In [2]: df=pd.read_csv(r"C:\Users\user\Downloads\used_cars_data.csv")
df
```

Out[2]:

	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Price
0	Mumbai	2010	72000	CNG	Manual	First	26.6 km/kg	998 CC	580000
1	Pune	2015	41000	Diesel	Manual	First	19.67 kmpl	1582 CC	1200000
2	Chennai	2011	46000	Petrol	Manual	First	18.2 kmpl	1199 CC	800000
3	Chennai	2012	87000	Diesel	Manual	First	20.77 kmpl	1248 CC	880000
4	Coimbatore	2013	40670	Diesel	Automatic	Second	15.2 kmpl	1968 CC	1400000
5
6	Hyderabad	2011	89411	Diesel	Manual	First	20.54 kmpl	1598 CC	1000000
7	Mumbai	2015	59000	Petrol	Automatic	First	17.21 kmpl	1197 CC	1000000
8	Kolkata	2012	28000	Diesel	Manual	First	23.08 kmpl	1461 CC	600000
9	Pune	2013	52262	Petrol	Automatic	Third	17.2 kmpl	1197 CC	1000000
10	Kochi	2014	72443	Diesel	Automatic	First	10.0 kmpl	2148 CC	1000000

```
In [3]: print('This Dataframe contains %d Rows and %d Columns'%(df.shape))
```

This Dataframe contains 7253 Rows and 14 Columns

```
In [4]: df.head()
```

Out[4]:

	S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second

```
In [5]: df.tail()
```

Out[5]:

	S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type
7248	7248	Volkswagen Vento Diesel Trendline	Hyderabad	2011	89411	Diesel	Manual	F
7249	7249	Volkswagen Polo GT TSI	Mumbai	2015	59000	Petrol	Automatic	F
7250	7250	Nissan Micra Diesel XV	Kolkata	2012	28000	Diesel	Manual	F
7251	7251	Volkswagen Polo GT TSI	Pune	2013	52262	Petrol	Automatic	Th
7252	7252	Mercedes-Benz E-Class 2009-2013 E 220 CDI Avan...	Kochi	2014	72443	Diesel	Automatic	F

In [6]: `df.describe()`

Out[6]:

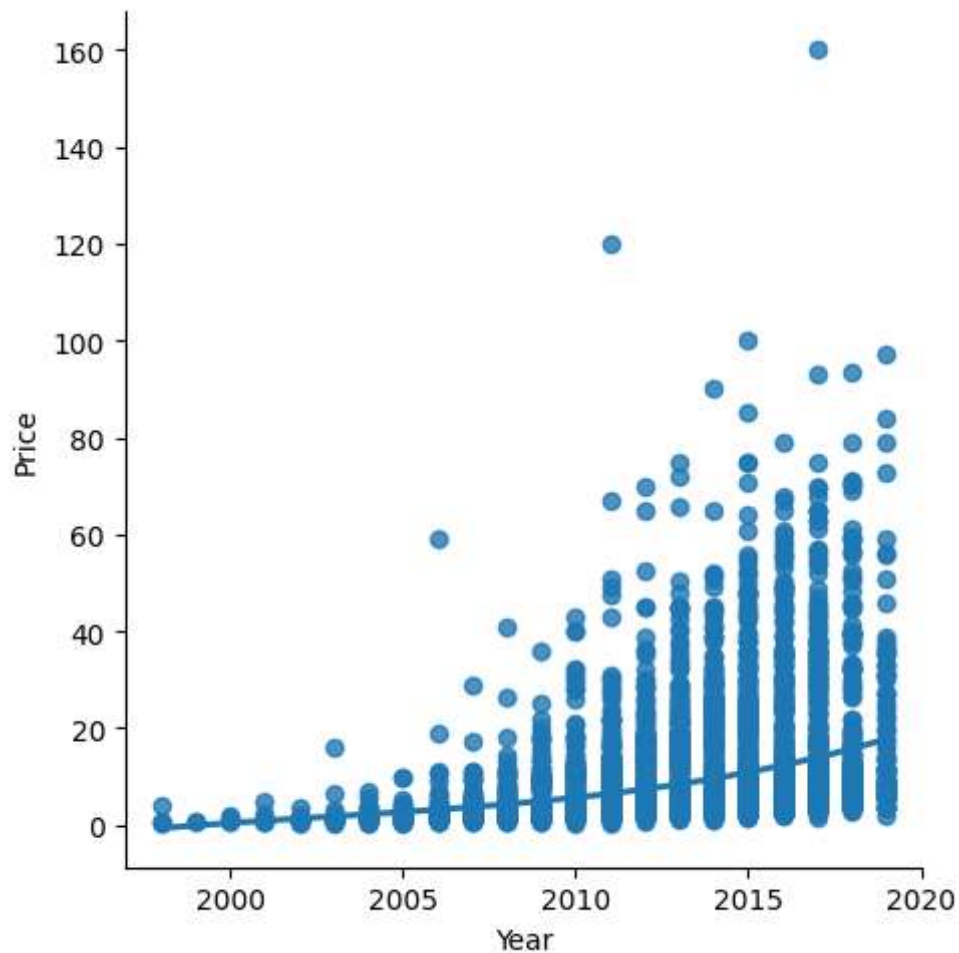
	S.No.	Year	Kilometers_Driven	Seats	Price
count	7253.000000	7253.000000	7.253000e+03	7200.000000	6019.000000
mean	3626.000000	2013.365366	5.869906e+04	5.279722	9.479468
std	2093.905084	3.254421	8.442772e+04	0.811660	11.187917
min	0.000000	1996.000000	1.710000e+02	0.000000	0.440000
25%	1813.000000	2011.000000	3.400000e+04	5.000000	3.500000
50%	3626.000000	2014.000000	5.341600e+04	5.000000	5.640000
75%	5439.000000	2016.000000	7.300000e+04	5.000000	9.950000
max	7252.000000	2019.000000	6.500000e+06	10.000000	160.000000

In [7]: `df.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7253 entries, 0 to 7252
Data columns (total 14 columns):
#   Column                Non-Null Count  Dtype
---  -
0   S.No.                 7253 non-null  int64
1   Name                  7253 non-null  object
2   Location              7253 non-null  object
3   Year                  7253 non-null  int64
4   Kilometers_Driven     7253 non-null  int64
5   Fuel_Type             7253 non-null  object
6   Transmission          7253 non-null  object
7   Owner_Type            7253 non-null  object
8   Mileage               7251 non-null  object
9   Engine                7207 non-null  object
10  Power                 7207 non-null  object
11  Seats                 7200 non-null  float64
12  New_Price             1006 non-null  object
13  Price                 6019 non-null  float64
dtypes: float64(2), int64(3), object(9)
memory usage: 793.4+ KB
```

```
In [10]: sns.lmplot(x="Year",y="Price", data = df, order = 3, ci = None)
```

```
Out[10]: <seaborn.axisgrid.FacetGrid at 0x1cbbbe680d0>
```



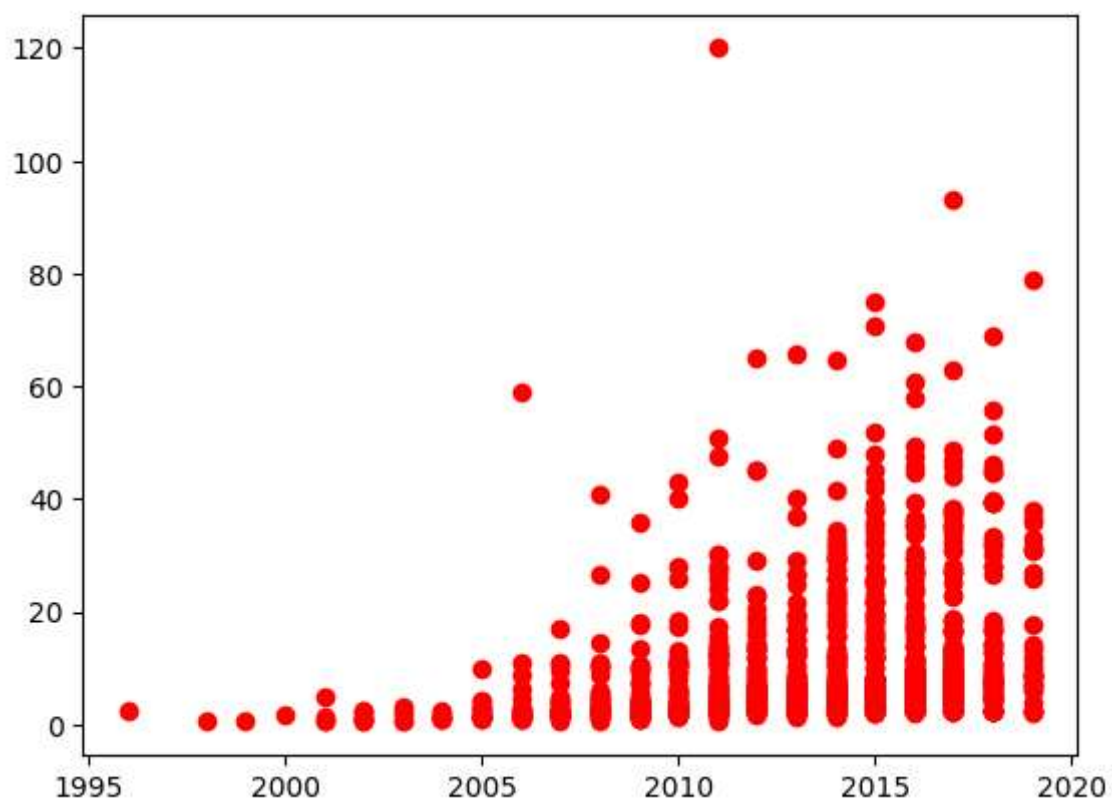
```
In [11]: df.fillna(method = 'ffill', inplace = True)
```

```
In [14]: X = np.array(df['Year']).reshape(-1, 1)
y = np.array(df['Price']).reshape(-1, 1)
```

```
In [16]: X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.25)
regr = LinearRegression()
regr.fit(X_train, y_train)
print(regr.score(X_test, y_test))
```

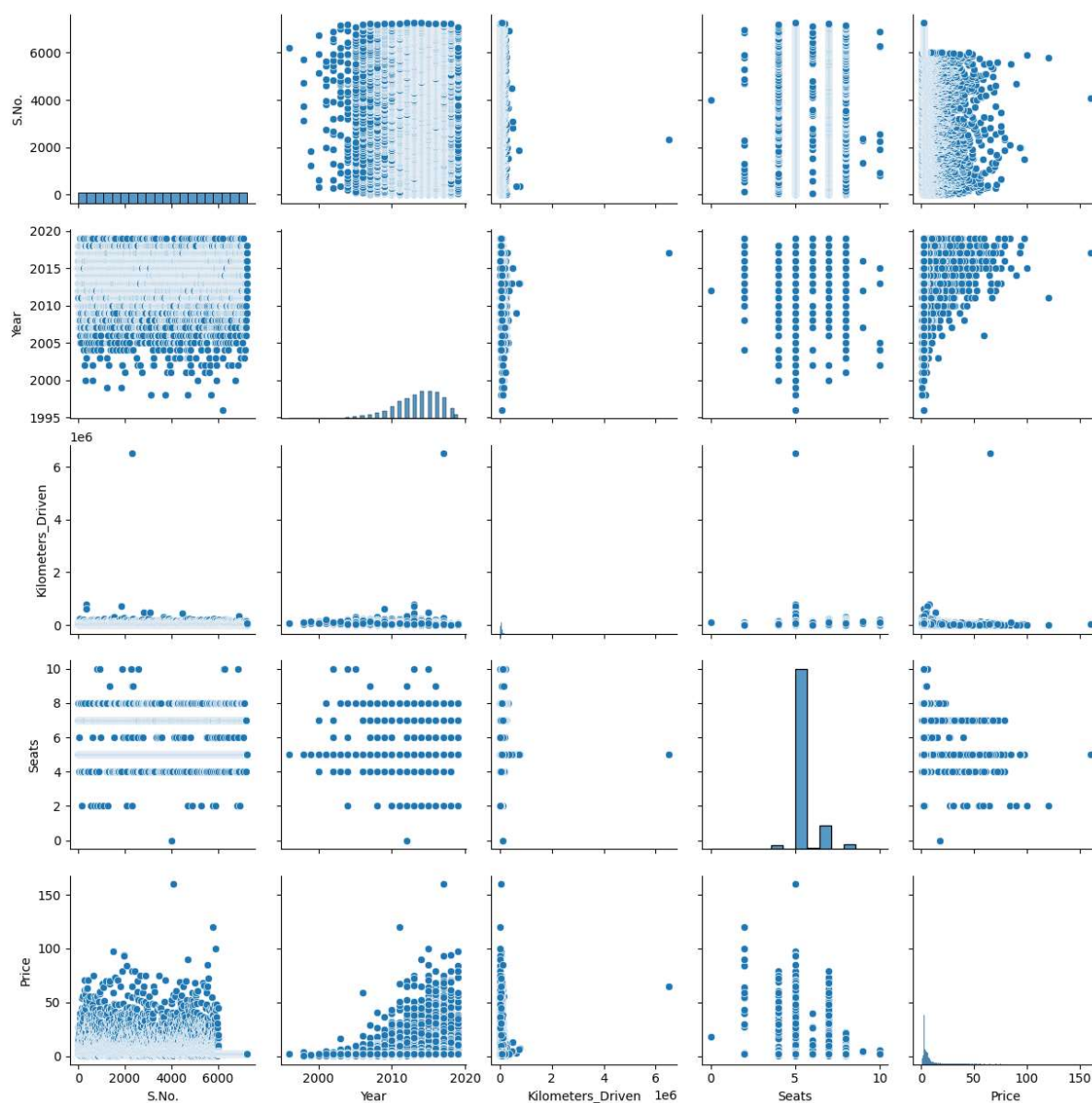
```
0.060481628690679745
```

```
In [18]: y_pred = regr.predict(X_test)
plt.scatter(X_test, y_test, color = 'r')
plt.show()
```



```
In [19]: sns.pairplot(df)
```

```
Out[19]: <seaborn.axisgrid.PairGrid at 0x1cbb7362110>
```



```
In [ ]:
```