In [49]:
```
import pandas as pd
import numpy as np
```

In [50]:
```
data=pd.read_csv(r"C:\Users\Param\Desktop\new used cars.csv")
```

In [51]:
```
data
```

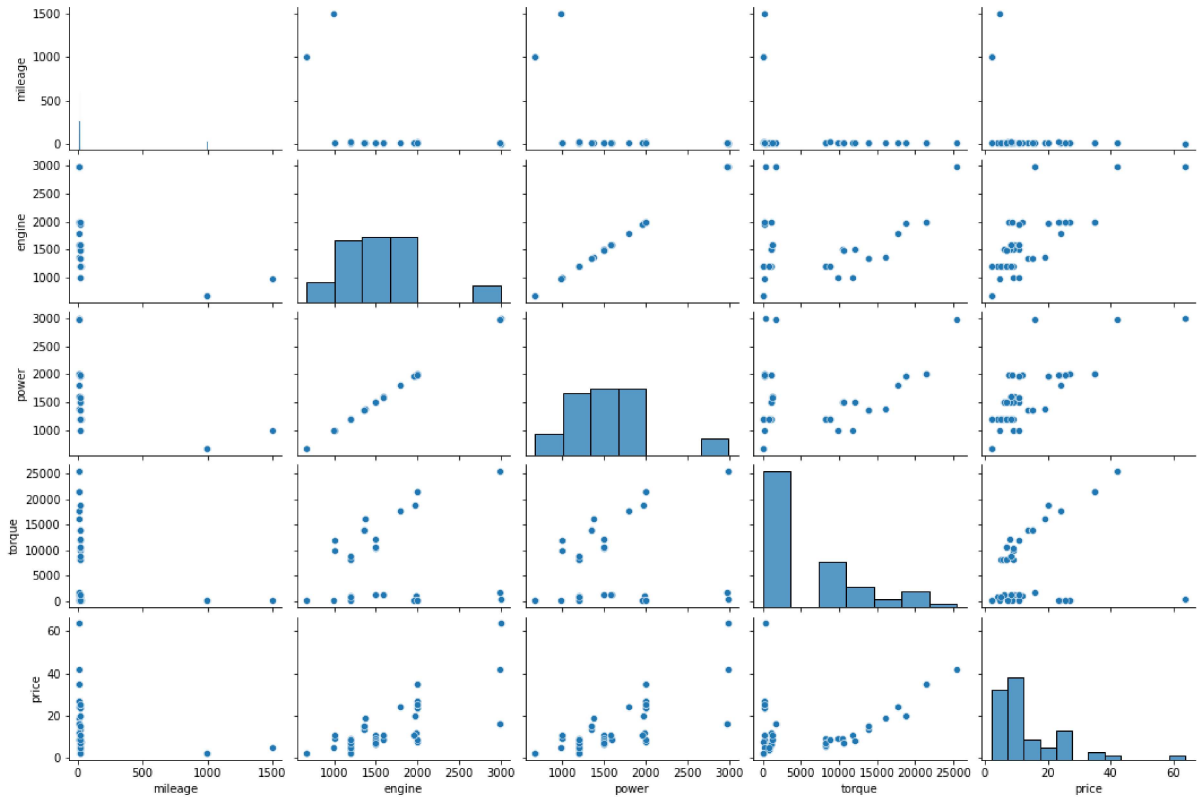
Out[51]:

	mileage	engine	power	torque	price
0	7.81	2996	2996	333	63.75
1	17.40	999	999	9863	8.99
2	20.68	1995	1995	188	23.75
3	16.50	1353	1353	13808	13.56
4	14.67	1798	1798	17746	24.00
5	18.70	1199	1199	887	5.45
6	18.90	1197	1197	8186	5.12
7	15.80	1591	1591	1213	9.30
8	13.50	2987	2987	25479	42.00
9	17.00	1198	1198	1085	8.02
10	17.40	1497	1497	1176	10.95
11	16.42	1498	1498	10455	8.99
12	18.88	1995	1995	184	7.45
13	18.15	998	998	11835	10.95
14	18.90	1197	1197	8186	5.78
15	21.00	1197	1197	8180	8.95
16	18.88	1995	1995	184	8.50
17	22.69	1995	1995	190	23.50
18	17.00	1497	1497	12136	7.95
19	14.10	1368	1368	16077	18.95
20	12.40	1996	1996	21501	35.00
21	16.10	1197	1197	85	2.09
22	18.00	1497	1497	1173	6.25
23	998.00	671	671	90	2.12
24	18.53	1968	1968	18774	20.00
25	15.29	1591	1591	1213	8.35
26	21.40	1197	1197	831	5.03
27	17.10	1496	1496	10594	6.95
28	12.55	2982	2982	1685	16.00
29	16.80	1353	1353	13808	15.22
30	13.50	1999	1999	177	26.95
31	18.60	1197	1197	8186	6.70
32	17.10	1956	1956	170	10.75
33	20.68	1995	1995	188	25.50
34	22.56	1197	1197	8850	8.38
35	17.60	1582	1582	1262	10.90

	mileage	engine	power	torque	price
36	17.80	1198	1198	867	3.95
37	26.49	1199	1199	72	7.25
38	1498.00	986	986	200	4.75
39	10.98	1984	1984	1144	11.95
40	12.40	1996	1996	21501	35.00
41	16.10	1197	1197	85	2.09
42	18.00	1497	1497	1173	6.25
43	998.00	671	671	90	2.12
44	18.53	1968	1968	18774	20.00
45	15.29	1591	1591	1213	8.35
46	21.40	1197	1197	831	5.03
47	17.10	1496	1496	10594	6.95
48	12.55	2982	2982	1685	16.00
49	16.80	1353	1353	13808	15.22
50	13.50	1999	1999	177	26.95
51	18.60	1197	1197	8186	6.70
52	17.10	1956	1956	170	10.75
53	20.68	1995	1995	188	25.50
54	22.56	1197	1197	8850	8.38
55	17.60	1582	1582	1262	10.90

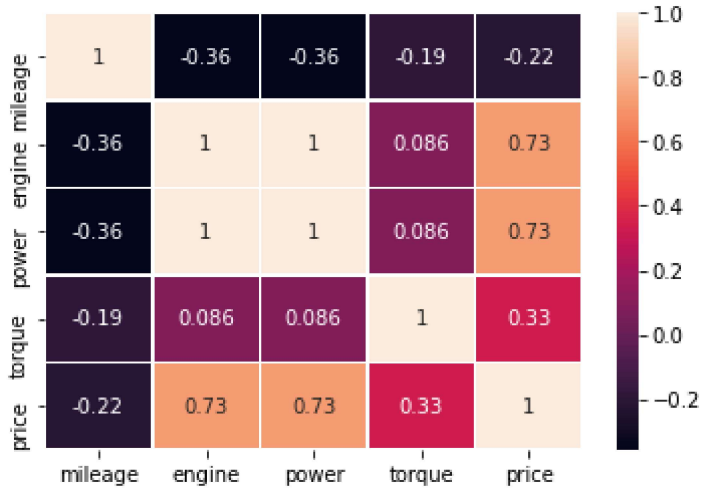
In [52]:
```
import seaborn as sns
sns.pairplot(data,height=2,aspect=1.5)
```

Out[52]:
```
<seaborn.axisgrid.PairGrid at 0x14c0a3c7700>
```



In [53]:
```
sns.heatmap(data.corr(),annot=True, linewidth=0.6)
```

Out[53]:
```
<AxesSubplot:>
```



In [54]:
```
X=data.drop(['price','power'],axis='columns')
print(X)
Y=data.drop(['mileage','engine','power','torque'], axis='columns')
print (Y)
```

```
37   7.25
38   4.75
39  11.95
40  35.00
41   2.09
42   6.25
43   2.12
44  20.00
45   8.35
46   5.03
47   6.95
48  16.00
49  15.22
50  26.95
51   6.70
52  10.75
53  25.50
54   8.38
55  10.90
```

In [55]:
```
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_stat
print(X_train)
print(X_test)
print(Y_train)
print(Y_test)
```

```
6    5.12
14   5.78
36   3.95
24  20.00
5    5.45
38   4.75
33  25.50
4   24.00
3   13.56
37   7.25
20  35.00
26   5.03
44  20.00
21   2.09
48  16.00
31   6.70
34   8.38
7    9.30
55  10.90
```

In [56]:
```
from sklearn.linear_model import LinearRegression
equation=LinearRegression()
```

In [57]:
```
equation.fit(X_train,Y_train)
```

Out[57]:
```
LinearRegression()
```

In [58]:
```
print('Intercept is', equation.intercept_)

print('Coefficient is', equation.coef_)
```

```
Intercept is [-11.58658639]
Coefficient is [[0.00312969 0.01296001 0.00058801]]
```

In [59]:
```
Y_test_predicted=equation.predict(X_test)
print(Y_test_predicted)
```

```
[[ 9.71329668]
 [14.46679707]
 [14.44389631]
 [13.91666756]
 [ 8.79820564]
 [14.08448174]
 [27.46184616]
 [14.12012633]
 [ 7.21446153]
 [ 8.56061502]
 [28.09023047]
 [ 4.63069821]
 [ 8.56050123]
 [ 8.36340477]
 [14.08448174]
 [13.91666756]
 [ 0.28592741]]
```

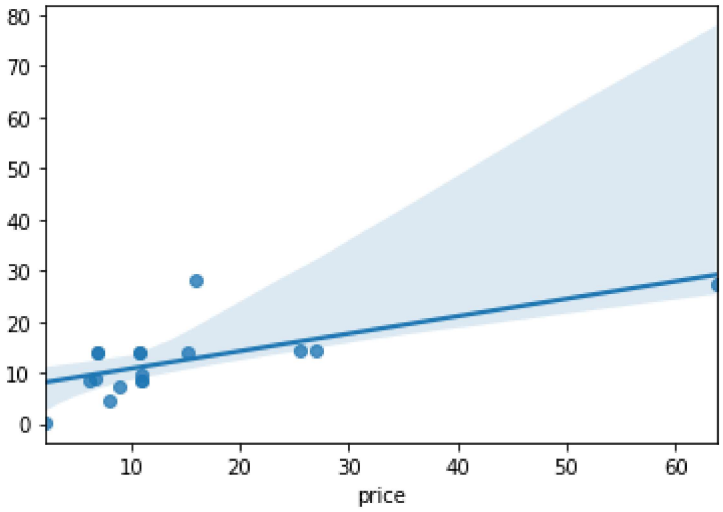
In [62]:
```
from sklearn import metrics
MAE=metrics.mean_absolute_error(Y_test,Y_test_predicted)
print(MAE)
r2=metrics.r2_score(Y_test,Y_test_predicted)
print(r2)
```

```
6.540611412301271
0.4116529335710165
```

In [61]:
```
import matplotlib.pyplot as plt

sns.regplot(x=Y_test, y=Y_test_predicted)
```

Out[61]: <AxesSubplot:xlabel='price'>



In [ ]: