ai-program-day-5

May 5, 2024

[77]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

[78]:
```
df=pd.read_csv('citizen_pci.csv')
```

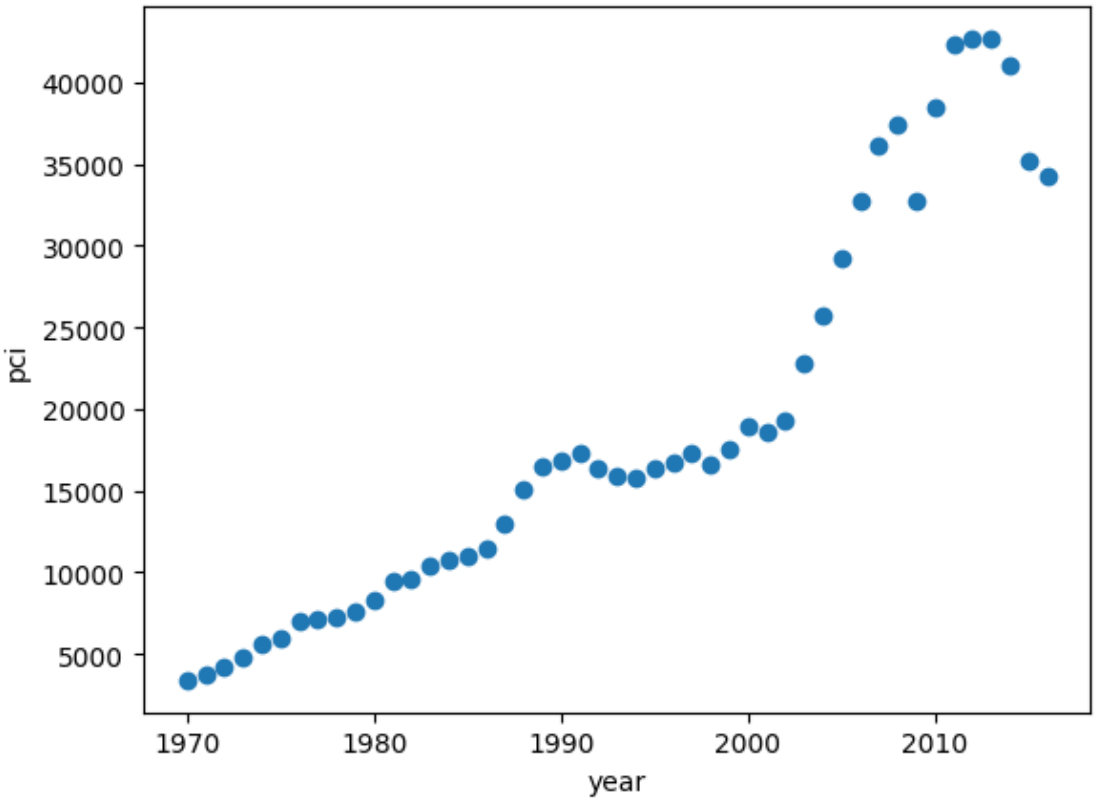
[79]:
```
df
```

[79]:

	year	pci
0	1970	3399.299037
1	1971	3768.297935
2	1972	4251.175484
3	1973	4804.463248
4	1974	5576.514583
5	1975	5998.144346
6	1976	7062.131392
7	1977	7100.126170
8	1978	7247.967035
9	1979	7602.912681
10	1980	8355.968120
11	1981	9434.390652
12	1982	9619.438377
13	1983	10416.536590
14	1984	10790.328720
15	1985	11018.955850
16	1986	11482.891530
17	1987	12974.806620
18	1988	15080.283450
19	1989	16426.725480
20	1990	16838.673200
21	1991	17266.097690
22	1992	16412.083090
23	1993	15875.586730
24	1994	15755.820270

```
25  1995  16369.317250
26  1996  16699.826680
27  1997  17310.757750
28  1998  16622.671870
29  1999  17581.024140
30  2000  18987.382410
31  2001  18601.397240
32  2002  19232.175560
33  2003  22739.426280
34  2004  25719.147150
35  2005  29198.055690
36  2006  32738.262900
37  2007  36144.481220
38  2008  37446.486090
39  2009  32755.176820
40  2010  38420.522890
41  2011  42334.711210
42  2012  42665.255970
43  2013  42676.468370
44  2014  41039.893600
45  2015  35175.188980
46  2016  34229.193630
```

[80]:
```
plt.scatter(df.year,df.pci)
plt.xlabel('year')
plt.ylabel('pci')
plt.show()
```



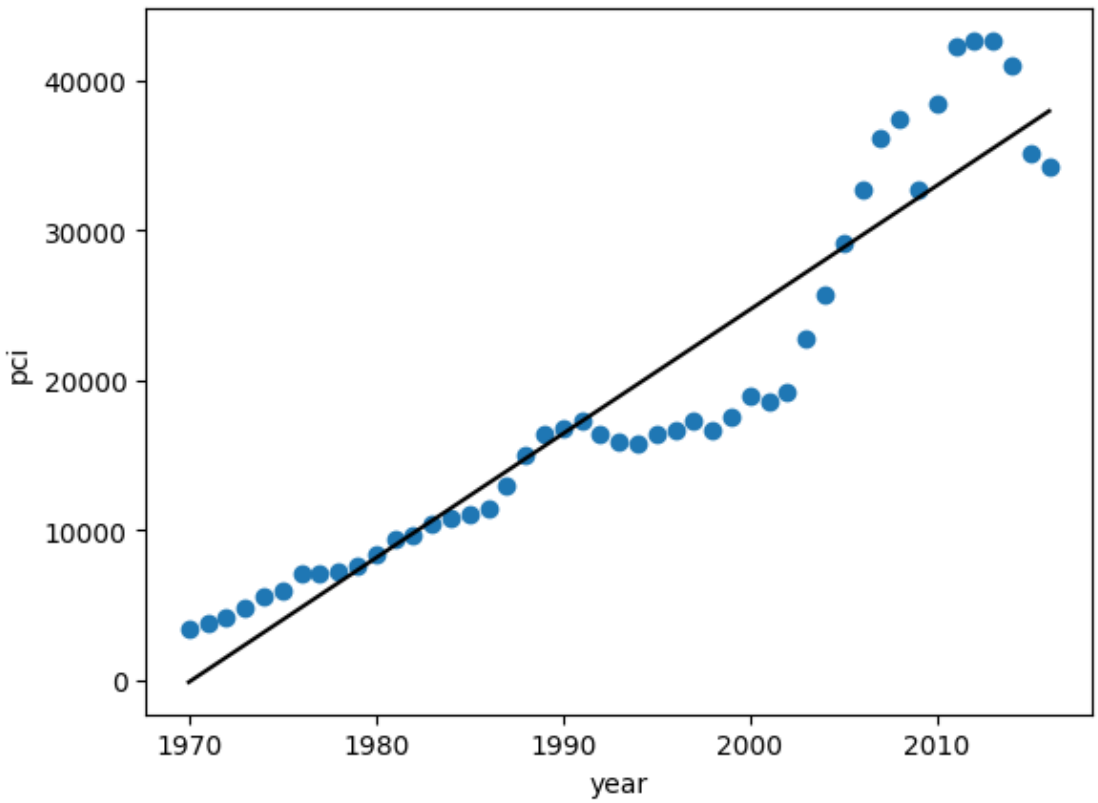
[81]:
```
import sklearn.linear_model as linear_model
reg=linear_model.LinearRegression()
reg.fit(df[['year']],df.pci)
```

[81]:
```
LinearRegression()
```

[82]:
```
reg.predict([[2030]])
```

[82]:
```
array([49573.34484664])
```

[83]:
```
plt.scatter(df.year,df.pci)
plt.plot(df.year,reg.predict(df[['year']]),color='black')
plt.xlabel('year')
plt.ylabel('pci')
plt.show()
```



[84]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

[85]:
```
df1=pd.read_csv('Salary_Data.csv')
```

[86]:
```
df1
```

[86]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891
5	2.9	56642
6	3.0	60150
7	3.2	54445
8	3.2	64445

```
9              3.7   57189
10             3.9   63218
11             4.0   55794
12             4.0   56957
13             4.1   57081
14             4.5   61111
15             4.9   67938
16             5.1   66029
17             5.3   83088
18             5.9   81363
19             6.0   93940
20             6.8   91738
21             7.1   98273
22             7.9  101302
23             8.2  113812
24             8.7  109431
25             9.0  105582
26             9.5  116969
27             9.6  112635
28            10.3  122391
29            10.5  121872
```

[87]:
```
x=df1.iloc[:,:-1].values
```

[88]:
```
print(x)
```

```
[[ 1.1]
 [ 1.3]
 [ 1.5]
 [ 2. ]
 [ 2.2]
 [ 2.9]
 [ 3. ]
 [ 3.2]
 [ 3.2]
 [ 3.7]
 [ 3.9]
 [ 4. ]
 [ 4. ]
 [ 4.1]
 [ 4.5]
 [ 4.9]
 [ 5.1]
 [ 5.3]
 [ 5.9]
 [ 6. ]
 [ 6.8]
```

```
[ 7.1]
[ 7.9]
[ 8.2]
[ 8.7]
[ 9. ]
[ 9.5]
[ 9.6]
[10.3]
[10.5]]
```

[89]:
```
df1.head()
```

[89]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891

[90]:
```
df1.iloc[0]
```

[90]:
```
YearsExperience        1.1
Salary             39343.0
Name: 0, dtype: float64
```

[91]:
```
df1.iloc[:10]
```

[91]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891
5	2.9	56642
6	3.0	60150
7	3.2	54445
8	3.2	64445
9	3.7	57189

[92]:
```
df1.iloc[0:]
```

[92]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891

```
5                2.9   56642
6                3.0   60150
7                3.2   54445
8                3.2   64445
9                3.7   57189
10               3.9   63218
11               4.0   55794
12               4.0   56957
13               4.1   57081
14               4.5   61111
15               4.9   67938
16               5.1   66029
17               5.3   83088
18               5.9   81363
19               6.0   93940
20               6.8   91738
21               7.1   98273
22               7.9  101302
23               8.2  113812
24               8.7  109431
25               9.0  105582
26               9.5  116969
27               9.6  112635
28              10.3  122391
29              10.5  121872
```

[93]:
```
df1.iloc[:10,1]
```

[93]:
```
0    39343
1    46205
2    37731
3    43525
4    39891
5    56642
6    60150
7    54445
8    64445
9    57189
Name: Salary, dtype: int64
```

[94]:
```
df1.iloc[:10,-1]
```

[94]:
```
0    39343
1    46205
2    37731
3    43525
4    39891
```

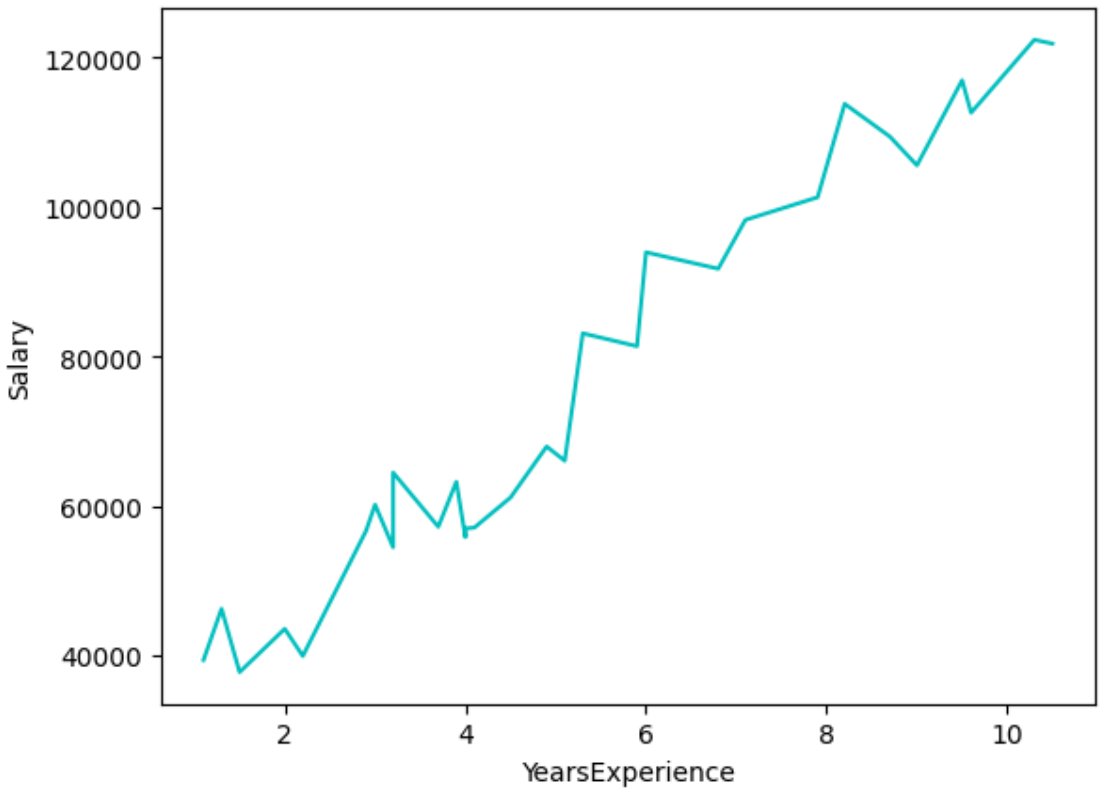
```
5    56642
6    60150
7    54445
8    64445
9    57189
Name: Salary, dtype: int64
```

[95]:
```
y=df1.iloc[:,1].values
```

[96]:
```
print(y)
```

```
[ 39343  46205  37731  43525  39891  56642  60150  54445  64445  57189
  63218  55794  56957  57081  61111  67938  66029  83088  81363  93940
  91738  98273 101302 113812 109431 105582 116969 112635 122391 121872]
```

[97]:
```
plt.plot(x,y,'c')
plt.xlabel("YearsExperience")
plt.ylabel("Salary")
plt.show()
```



[98]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=1/3,random_state=4)
```

[99]:
```
print(x.shape)
print(y.shape)
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(30, 1)
(30,)
(20, 1)
(10, 1)
(20,)
(10,)
```

[100]:
```
print("X Shape:",x.shape)
print("y Shape:",y.shape)
print("X Train:",x_train.shape)
print("X Test:",x_test.shape)
print("Y Train:",y_train.shape)
print("Y Test:",y_test.shape)
```

```
X Shape: (30, 1)
y Shape: (30,)
X Train: (20, 1)
X Test: (10, 1)
Y Train: (20,)
Y Test: (10,)
```

[101]:
```
from sklearn.linear_model import LinearRegression
regressor=LinearRegression()
regressor.fit(x_train,y_train)
print(regressor.intercept_)
print(regressor.coef_)
print(regressor)
```

```
24839.263212221318
[9569.34042114]
LinearRegression()
```

[102]:
```
y_pred=regressor.predict(x_test)
y_pred
```

[102]:
```
array([ 63116.62489678,  92781.58020231, 123403.46954996,  71729.03127581,
        89910.77807597, 110963.32700248,  75556.76744426, 125317.33763419,
```

```
       82255.30573906,  35365.53767547])
```

[103]:
```
df1_x_test=pd.DataFrame(x_test)
df1_y_test=pd.DataFrame(y_test)
df1_y_pred=pd.DataFrame(y_pred)
df1_diff=df1_y_test-df1_y_pred
y_test_pred=pd.concat([df1_x_test,df1_y_test,df1_y_pred],axis=1)
y_test_pred['Difference']=df1_diff
```

[104]:
```
y_test_pred
```

[104]:

	0	0	0	Difference
0	4.0	55794	63116.624897	-7322.624897
1	7.1	98273	92781.580202	5491.419798
2	10.3	122391	123403.469550	-1012.469550
3	4.9	67938	71729.031276	-3791.031276
4	6.8	91738	89910.778076	1827.221924
5	9.0	105582	110963.327002	-5381.327002
6	5.3	83088	75556.767444	7531.232556
7	10.5	121872	125317.337634	-3445.337634
8	6.0	93940	82255.305739	11684.694261
9	1.1	39343	35365.537675	3977.462325

[105]:
```
df1_x_test=pd.DataFrame(x_test,columns=['Experience'])
df1_y_test=pd.DataFrame(y_test,columns=['Salary'])
df1_y_pred=pd.DataFrame(y_pred,columns=['Prediction'])
df1_diff=df1_y_test-df1_y_pred
y_test_pred=pd.concat([df1_x_test,df1_y_test,df1_y_pred],axis=1)
y_test_pred['Difference']=y_test_pred['Salary']-y_test_pred['Prediction']
y_test_pred
```

[105]:

	Experience	Salary	Prediction	Difference
0	4.0	55794	63116.624897	-7322.624897
1	7.1	98273	92781.580202	5491.419798
2	10.3	122391	123403.469550	-1012.469550
3	4.9	67938	71729.031276	-3791.031276
4	6.8	91738	89910.778076	1827.221924
5	9.0	105582	110963.327002	-5381.327002
6	5.3	83088	75556.767444	7531.232556
7	10.5	121872	125317.337634	-3445.337634
8	6.0	93940	82255.305739	11684.694261
9	1.1	39343	35365.537675	3977.462325

[106]:
```
from sklearn.metrics import r2_score
r2_score(y_test,y_pred)
```

[106]:
```
0.9468080828036048
```

[107]:
```
y_pred_f=regressor.predict(x_test)
y_pred_f
```

[107]:
```
array([ 63116.62489678,  92781.58020231, 123403.46954996,  71729.03127581,
        89910.77807597, 110963.32700248,  75556.76744426, 125317.33763419,
        82255.30573906,  35365.53767547])
```

[108]:
```
y_pred_f=regressor.predict(x)
y_pred_f
```

[108]:
```
array([ 35365.53767547,  37279.4057597 ,  39193.27384393,  43977.9440545 ,
        45891.81213873,  52590.35043353,  53547.28447564,  55461.15255987,
        55461.15255987,  60245.82277044,  62159.69085467,  63116.62489678,
        63116.62489678,  64073.55893889,  67901.29510735,  71729.03127581,
        73642.89936003,  75556.76744426,  81298.37169694,  82255.30573906,
        89910.77807597,  92781.58020231, 100437.05253922, 103307.85466557,
       108092.52487614, 110963.32700248, 115747.99721305, 116704.93125516,
       123403.46954996, 125317.33763419])
```

[109]:
```
y_pred_final=pd.DataFrame(y_pred_f,columns=['prediction'])
y_pred_final
```

[109]:

	prediction
0	35365.537675
1	37279.405760
2	39193.273844
3	43977.944055
4	45891.812139
5	52590.350434
6	53547.284476
7	55461.152560
8	55461.152560
9	60245.822770
10	62159.690855
11	63116.624897
12	63116.624897
13	64073.558939
14	67901.295107
15	71729.031276
16	73642.899360
17	75556.767444
18	81298.371697
19	82255.305739
20	89910.778076
21	92781.580202
22	100437.052539
23	103307.854666

```
24  108092.524876
25  110963.327002
26  115747.997213
27  116704.931255
28  123403.469550
29  125317.337634
```

[110]:
```
result=pd.concat([df1,y_pred_final],axis=1)
result
```

[110]:

	YearsExperience	Salary	prediction
0	1.1	39343	35365.537675
1	1.3	46205	37279.405760
2	1.5	37731	39193.273844
3	2.0	43525	43977.944055
4	2.2	39891	45891.812139
5	2.9	56642	52590.350434
6	3.0	60150	53547.284476
7	3.2	54445	55461.152560
8	3.2	64445	55461.152560
9	3.7	57189	60245.822770
10	3.9	63218	62159.690855
11	4.0	55794	63116.624897
12	4.0	56957	63116.624897
13	4.1	57081	64073.558939
14	4.5	61111	67901.295107
15	4.9	67938	71729.031276
16	5.1	66029	73642.899360
17	5.3	83088	75556.767444
18	5.9	81363	81298.371697
19	6.0	93940	82255.305739
20	6.8	91738	89910.778076
21	7.1	98273	92781.580202
22	7.9	101302	100437.052539
23	8.2	113812	103307.854666
24	8.7	109431	108092.524876
25	9.0	105582	110963.327002
26	9.5	116969	115747.997213
27	9.6	112635	116704.931255
28	10.3	122391	123403.469550
29	10.5	121872	125317.337634

[111]:
```
acc=r2_score(y,y_pred_f)
acc
```

[111]: 0.9566641936753054

[112]:
```
regressor.predict([[5]])
```

[112]:
```
array([72685.96531792])
```

[140]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import r2_score
from sklearn.model_selection import train_test_split
df2=pd.read_csv("50_Startups.csv")
df2
```

[140]:

	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349.20	136897.80	471784.10	New York	192261.83
1	162597.70	151377.59	443898.53	California	191792.06
2	153441.51	101145.55	407934.54	Florida	191050.39
3	144372.41	118671.85	383199.62	New York	182901.99
4	142107.34	91391.77	366168.42	California	166187.94
5	131876.90	99814.71	362861.36	New York	156991.12
6	134615.46	147198.87	127716.82	California	156122.51
7	130298.13	145530.06	323876.68	New York	155752.60
8	120542.52	148718.95	311613.29	New York	152211.77
9	123334.88	108679.17	304981.62	California	149759.96
10	101913.08	110594.11	229160.95	California	146121.95
11	100671.96	91790.61	249744.55	Florida	144259.40
12	93863.75	127320.38	249839.44	California	141585.52
13	91992.39	135495.07	252664.93	California	134307.35
14	119943.24	156547.42	256512.92	Florida	132602.65
15	114523.61	122616.84	261776.23	New York	129917.04
16	78013.11	121597.55	264346.06	California	126992.93
17	94657.16	145077.58	282574.31	New York	125370.37
18	91749.16	114175.79	294919.57	New York	124266.90
19	86419.70	153514.11	0.00	Florida	122776.86
20	76253.86	113867.30	298664.47	California	118474.03
21	78389.47	153773.43	299737.29	New York	111313.02
22	73994.56	122782.75	303319.26	California	110352.25
23	67532.53	105751.03	304768.73	California	108733.99
24	77044.01	99281.34	140574.81	New York	108552.04
25	64664.71	139553.16	137962.62	California	107404.34
26	75328.87	144135.98	134050.07	New York	105733.54
27	72107.60	127864.55	353183.81	New York	105008.31
28	66051.52	182645.56	118148.20	Florida	103282.38
29	65605.48	153032.06	107138.38	New York	101004.64
30	61994.48	115641.28	91131.24	New York	99937.59
31	61136.38	152701.92	88218.23	New York	97483.56
32	63408.86	129219.61	46085.25	California	97427.84
33	55493.95	103057.49	214634.81	New York	96778.92

34	46426.07	157693.92	210797.67	Florida	96712.80
35	46014.02	85047.44	205517.64	New York	96479.51
36	28663.76	127056.21	201126.82	New York	90708.19
37	44069.95	51283.14	197029.42	California	89949.14
38	20229.59	65947.93	185265.10	New York	81229.06
39	38558.51	82982.09	174999.30	California	81005.76
40	28754.33	118546.05	172795.67	Florida	78239.91
41	27892.92	84710.77	164470.71	California	77798.83
42	23640.93	96189.63	148001.11	California	71498.49
43	15505.73	127382.30	35534.17	New York	69758.98
44	22177.74	154806.14	28334.72	California	65200.33
45	1000.23	124153.04	1903.93	New York	64926.08
46	1315.46	115816.21	297114.46	Florida	49490.75
47	0.00	135426.92	0.00	California	42559.73
48	542.05	51743.15	0.00	New York	35673.41
49	0.00	116983.80	45173.06	California	14681.40

[141]:
```
x=df2.iloc[:,:-1].values
y=df2.iloc[:,4].values
```

[142]:
```
print(x,y)
```

```
[[165349.2 136897.8 471784.1 'New York']
 [162597.7 151377.59 443898.53 'California']
 [153441.51 101145.55 407934.54 'Florida']
 [144372.41 118671.85 383199.62 'New York']
 [142107.34 91391.77 366168.42 'California']
 [131876.9 99814.71 362861.36 'New York']
 [134615.46 147198.87 127716.82 'California']
 [130298.13 145530.06 323876.68 'New York']
 [120542.52 148718.95 311613.29 'New York']
 [123334.88 108679.17 304981.62 'California']
 [101913.08 110594.11 229160.95 'California']
 [100671.96 91790.61 249744.55 'Florida']
 [93863.75 127320.38 249839.44 'California']
 [91992.39 135495.07 252664.93 'California']
 [119943.24 156547.42 256512.92 'Florida']
 [114523.61 122616.84 261776.23 'New York']
 [78013.11 121597.55 264346.06 'California']
 [94657.16 145077.58 282574.31 'New York']
 [91749.16 114175.79 294919.57 'New York']
 [86419.7 153514.11 0.0 'Florida']
 [76253.86 113867.3 298664.47 'California']
 [78389.47 153773.43 299737.29 'New York']
 [73994.56 122782.75 303319.26 'California']
 [67532.53 105751.03 304768.73 'California']
 [77044.01 99281.34 140574.81 'New York']
```

```
 [64664.71 139553.16 137962.62 'California']
 [75328.87 144135.98 134050.07 'New York']
 [72107.6 127864.55 353183.81 'New York']
 [66051.52 182645.56 118148.2 'Florida']
 [65605.48 153032.06 107138.38 'New York']
 [61994.48 115641.28 91131.24 'New York']
 [61136.38 152701.92 88218.23 'New York']
 [63408.86 129219.61 46085.25 'California']
 [55493.95 103057.49 214634.81 'New York']
 [46426.07 157693.92 210797.67 'Florida']
 [46014.02 85047.44 205517.64 'New York']
 [28663.76 127056.21 201126.82 'New York']
 [44069.95 51283.14 197029.42 'California']
 [20229.59 65947.93 185265.1 'New York']
 [38558.51 82982.09 174999.3 'California']
 [28754.33 118546.05 172795.67 'Florida']
 [27892.92 84710.77 164470.71 'California']
 [23640.93 96189.63 148001.11 'California']
 [15505.73 127382.3 35534.17 'New York']
 [22177.74 154806.14 28334.72 'California']
 [1000.23 124153.04 1903.93 'New York']
 [1315.46 115816.21 297114.46 'Florida']
 [0.0 135426.92 0.0 'California']
 [542.05 51743.15 0.0 'New York']
 [0.0 116983.8 45173.06 'California']] [192261.83 191792.06 191050.39 182901.99
166187.94 156991.12 156122.51
 155752.6  152211.77 149759.96 146121.95 144259.4  141585.52 134307.35
 132602.65 129917.04 126992.93 125370.37 124266.9  122776.86 118474.03
 111313.02 110352.25 108733.99 108552.04 107404.34 105733.54 105008.31
 103282.38 101004.64  99937.59  97483.56  97427.84  96778.92  96712.8
  96479.51  90708.19  89949.14  81229.06  81005.76  78239.91  77798.83
  71498.49  69758.98  65200.33  64926.08  49490.75  42559.73  35673.41
  14681.4 ]
```

[143]:
```
from sklearn.preprocessing import LabelEncoder,OneHotEncoder
Label=LabelEncoder()
x[:,3]=Label.fit_transform(x[:,3])
```

[144]:
```
print(x)
```

```
[[165349.2 136897.8 471784.1 2]
 [162597.7 151377.59 443898.53 0]
 [153441.51 101145.55 407934.54 1]
 [144372.41 118671.85 383199.62 2]
 [142107.34 91391.77 366168.42 0]
 [131876.9 99814.71 362861.36 2]
 [134615.46 147198.87 127716.82 0]
 [130298.13 145530.06 323876.68 2]
```

```
[120542.52 148718.95 311613.29 2]
[123334.88 108679.17 304981.62 0]
[101913.08 110594.11 229160.95 0]
[100671.96 91790.61 249744.55 1]
[93863.75 127320.38 249839.44 0]
[91992.39 135495.07 252664.93 0]
[119943.24 156547.42 256512.92 1]
[114523.61 122616.84 261776.23 2]
[78013.11 121597.55 264346.06 0]
[94657.16 145077.58 282574.31 2]
[91749.16 114175.79 294919.57 2]
[86419.7 153514.11 0.0 1]
[76253.86 113867.3 298664.47 0]
[78389.47 153773.43 299737.29 2]
[73994.56 122782.75 303319.26 0]
[67532.53 105751.03 304768.73 0]
[77044.01 99281.34 140574.81 2]
[64664.71 139553.16 137962.62 0]
[75328.87 144135.98 134050.07 2]
[72107.6 127864.55 353183.81 2]
[66051.52 182645.56 118148.2 1]
[65605.48 153032.06 107138.38 2]
[61994.48 115641.28 91131.24 2]
[61136.38 152701.92 88218.23 2]
[63408.86 129219.61 46085.25 0]
[55493.95 103057.49 214634.81 2]
[46426.07 157693.92 210797.67 1]
[46014.02 85047.44 205517.64 2]
[28663.76 127056.21 201126.82 2]
[44069.95 51283.14 197029.42 0]
[20229.59 65947.93 185265.1 2]
[38558.51 82982.09 174999.3 0]
[28754.33 118546.05 172795.67 1]
[27892.92 84710.77 164470.71 0]
[23640.93 96189.63 148001.11 0]
[15505.73 127382.3 35534.17 2]
[22177.74 154806.14 28334.72 0]
[1000.23 124153.04 1903.93 2]
[1315.46 115816.21 297114.46 1]
[0.0 135426.92 0.0 0]
[542.05 51743.15 0.0 2]
[0.0 116983.8 45173.06 0]]
```

[145]:
```
from sklearn.compose import ColumnTransformer
ct=ColumnTransformer([('encoder',OneHotEncoder(),[3])],remainder='passthrough')
x=np.array(ct.fit_transform(x))
print(x)
```

```
[[0.0 0.0 1.0 165349.2 136897.8 471784.1]
 [1.0 0.0 0.0 162597.7 151377.59 443898.53]
 [0.0 1.0 0.0 153441.51 101145.55 407934.54]
 [0.0 0.0 1.0 144372.41 118671.85 383199.62]
 [1.0 0.0 0.0 142107.34 91391.77 366168.42]
 [0.0 0.0 1.0 131876.9 99814.71 362861.36]
 [1.0 0.0 0.0 134615.46 147198.87 127716.82]
 [0.0 0.0 1.0 130298.13 145530.06 323876.68]
 [0.0 0.0 1.0 120542.52 148718.95 311613.29]
 [1.0 0.0 0.0 123334.88 108679.17 304981.62]
 [1.0 0.0 0.0 101913.08 110594.11 229160.95]
 [0.0 1.0 0.0 100671.96 91790.61 249744.55]
 [1.0 0.0 0.0 93863.75 127320.38 249839.44]
 [1.0 0.0 0.0 91992.39 135495.07 252664.93]
 [0.0 1.0 0.0 119943.24 156547.42 256512.92]
 [0.0 0.0 1.0 114523.61 122616.84 261776.23]
 [1.0 0.0 0.0 78013.11 121597.55 264346.06]
 [0.0 0.0 1.0 94657.16 145077.58 282574.31]
 [0.0 0.0 1.0 91749.16 114175.79 294919.57]
 [0.0 1.0 0.0 86419.7 153514.11 0.0]
 [1.0 0.0 0.0 76253.86 113867.3 298664.47]
 [0.0 0.0 1.0 78389.47 153773.43 299737.29]
 [1.0 0.0 0.0 73994.56 122782.75 303319.26]
 [1.0 0.0 0.0 67532.53 105751.03 304768.73]
 [0.0 0.0 1.0 77044.01 99281.34 140574.81]
 [1.0 0.0 0.0 64664.71 139553.16 137962.62]
 [0.0 0.0 1.0 75328.87 144135.98 134050.07]
 [0.0 0.0 1.0 72107.6 127864.55 353183.81]
 [0.0 1.0 0.0 66051.52 182645.56 118148.2]
 [0.0 0.0 1.0 65605.48 153032.06 107138.38]
 [0.0 0.0 1.0 61994.48 115641.28 91131.24]
 [0.0 0.0 1.0 61136.38 152701.92 88218.23]
 [1.0 0.0 0.0 63408.86 129219.61 46085.25]
 [0.0 0.0 1.0 55493.95 103057.49 214634.81]
 [0.0 1.0 0.0 46426.07 157693.92 210797.67]
 [0.0 0.0 1.0 46014.02 85047.44 205517.64]
 [0.0 0.0 1.0 28663.76 127056.21 201126.82]
 [1.0 0.0 0.0 44069.95 51283.14 197029.42]
 [0.0 0.0 1.0 20229.59 65947.93 185265.1]
 [1.0 0.0 0.0 38558.51 82982.09 174999.3]
 [0.0 1.0 0.0 28754.33 118546.05 172795.67]
 [1.0 0.0 0.0 27892.92 84710.77 164470.71]
 [1.0 0.0 0.0 23640.93 96189.63 148001.11]
 [0.0 0.0 1.0 15505.73 127382.3 35534.17]
 [1.0 0.0 0.0 22177.74 154806.14 28334.72]
 [0.0 0.0 1.0 1000.23 124153.04 1903.93]
 [0.0 1.0 0.0 1315.46 115816.21 297114.46]
 [1.0 0.0 0.0 0.0 135426.92 0.0]
```

```
 [0.0 0.0 1.0 542.05 51743.15 0.0]
 [1.0 0.0 0.0 0.0 116983.8 45173.06]]
```

[146]:
```
print(ord('a'))
print(chr(97))
```

```
97
a
```

[147]:
```
message="Encrypt this message using python!"
encrypted_message=""
for i in message.lower():
    if i.isalpha():
        encrypted_message+=chr(219-ord(i))
    elif i==" ":
        encrypted_message+="**"
    else:
        encrypted_message+=i
print(encrypted_message)
```

```
vmxibkg**gsrh**nvhhztv**fhrmt**kbgslm!
```

[148]:
```
ord('z')
```

[148]: 122

[149]:
```
ord('a')
```

[149]: 97

[123]:
```
print(pd.DataFrame(x).head())
```

```
     0    1    2          3          4          5
0  0.0  0.0  1.0   165349.2   136897.8   471784.1
1  1.0  0.0  0.0   162597.7  151377.59  443898.53
2  0.0  1.0  0.0  153441.51  101145.55  407934.54
3  0.0  0.0  1.0  144372.41  118671.85  383199.62
4  1.0  0.0  0.0  142107.34   91391.77  366168.42
```

[124]:
```
from sklearn.model_selection import train_test_split
train_test_split(x,y)
```

[124]:
```
[array([[0.0, 0.0, 1.0, 91749.16, 114175.79, 294919.57],
        [0.0, 0.0, 1.0, 78389.47, 153773.43, 299737.29],
        [0.0, 0.0, 1.0, 65605.48, 153032.06, 107138.38],
        [0.0, 0.0, 1.0, 1000.23, 124153.04, 1903.93],
        [0.0, 1.0, 0.0, 86419.7, 153514.11, 0.0],
        [1.0, 0.0, 0.0, 23640.93, 96189.63, 148001.11],
```

```
       [1.0, 0.0, 0.0, 63408.86, 129219.61, 46085.25],
       [0.0, 0.0, 1.0, 131876.9, 99814.71, 362861.36],
       [0.0, 1.0, 0.0, 153441.51, 101145.55, 407934.54],
       [0.0, 0.0, 1.0, 144372.41, 118671.85, 383199.62],
       [0.0, 0.0, 1.0, 75328.87, 144135.98, 134050.07],
       [1.0, 0.0, 0.0, 67532.53, 105751.03, 304768.73],
       [0.0, 0.0, 1.0, 72107.6, 127864.55, 353183.81],
       [0.0, 0.0, 1.0, 165349.2, 136897.8, 471784.1],
       [0.0, 0.0, 1.0, 542.05, 51743.15, 0.0],
       [0.0, 1.0, 0.0, 28754.33, 118546.05, 172795.67],
       [0.0, 0.0, 1.0, 114523.61, 122616.84, 261776.23],
       [0.0, 0.0, 1.0, 46014.02, 85047.44, 205517.64],
       [1.0, 0.0, 0.0, 64664.71, 139553.16, 137962.62],
       [0.0, 0.0, 1.0, 28663.76, 127056.21, 201126.82],
       [0.0, 1.0, 0.0, 1315.46, 115816.21, 297114.46],
       [0.0, 1.0, 0.0, 46426.07, 157693.92, 210797.67],
       [0.0, 1.0, 0.0, 119943.24, 156547.42, 256512.92],
       [1.0, 0.0, 0.0, 101913.08, 110594.11, 229160.95],
       [0.0, 0.0, 1.0, 120542.52, 148718.95, 311613.29],
       [0.0, 0.0, 1.0, 61994.48, 115641.28, 91131.24],
       [0.0, 0.0, 1.0, 55493.95, 103057.49, 214634.81],
       [1.0, 0.0, 0.0, 162597.7, 151377.59, 443898.53],
       [1.0, 0.0, 0.0, 123334.88, 108679.17, 304981.62],
       [1.0, 0.0, 0.0, 0.0, 135426.92, 0.0],
       [0.0, 0.0, 1.0, 20229.59, 65947.93, 185265.1],
       [0.0, 1.0, 0.0, 66051.52, 182645.56, 118148.2],
       [1.0, 0.0, 0.0, 93863.75, 127320.38, 249839.44],
       [0.0, 0.0, 1.0, 130298.13, 145530.06, 323876.68],
       [1.0, 0.0, 0.0, 0.0, 116983.8, 45173.06],
       [1.0, 0.0, 0.0, 73994.56, 122782.75, 303319.26],
       [1.0, 0.0, 0.0, 142107.34, 91391.77, 366168.42]], dtype=object),
array([[0.0, 0.0, 1.0, 61136.38, 152701.92, 88218.23],
       [1.0, 0.0, 0.0, 76253.86, 113867.3, 298664.47],
       [1.0, 0.0, 0.0, 78013.11, 121597.55, 264346.06],
       [1.0, 0.0, 0.0, 27892.92, 84710.77, 164470.71],
       [1.0, 0.0, 0.0, 134615.46, 147198.87, 127716.82],
       [1.0, 0.0, 0.0, 22177.74, 154806.14, 28334.72],
       [1.0, 0.0, 0.0, 44069.95, 51283.14, 197029.42],
       [0.0, 0.0, 1.0, 15505.73, 127382.3, 35534.17],
       [0.0, 0.0, 1.0, 77044.01, 99281.34, 140574.81],
       [1.0, 0.0, 0.0, 91992.39, 135495.07, 252664.93],
       [1.0, 0.0, 0.0, 38558.51, 82982.09, 174999.3],
       [0.0, 1.0, 0.0, 100671.96, 91790.61, 249744.55],
       [0.0, 0.0, 1.0, 94657.16, 145077.58, 282574.31]], dtype=object),
array([124266.9 , 111313.02, 101004.64,  64926.08, 122776.86,  71498.49,
        97427.84, 156991.12, 191050.39, 182901.99, 105733.54, 108733.99,
       105008.31, 192261.83,  35673.41,  78239.91, 129917.04,  96479.51,
```

```
        107404.34,  90708.19,  49490.75,  96712.8 , 132602.65, 146121.95,
        152211.77,  99937.59,  96778.92, 191792.06, 149759.96,  42559.73,
         81229.06, 103282.38, 141585.52, 155752.6 ,  14681.4 , 110352.25,
        166187.94]),
 array([ 97483.56, 118474.03, 126992.93,  77798.83, 156122.51,  65200.33,
         89949.14,  69758.98, 108552.04, 134307.35,  81005.76, 144259.4 ,
        125370.37])]
```

[125]:
```
from sklearn.model_selection import train_test_split
train_test_split(x,y,test_size=0.2,random_state=14)
```

[125]:
```
[array([[0.0, 0.0, 1.0, 28663.76, 127056.21, 201126.82],
        [1.0, 0.0, 0.0, 22177.74, 154806.14, 28334.72],
        [0.0, 0.0, 1.0, 94657.16, 145077.58, 282574.31],
        [0.0, 1.0, 0.0, 100671.96, 91790.61, 249744.55],
        [1.0, 0.0, 0.0, 38558.51, 82982.09, 174999.3],
        [0.0, 0.0, 1.0, 65605.48, 153032.06, 107138.38],
        [0.0, 0.0, 1.0, 61136.38, 152701.92, 88218.23],
        [0.0, 0.0, 1.0, 91749.16, 114175.79, 294919.57],
        [0.0, 1.0, 0.0, 46426.07, 157693.92, 210797.67],
        [0.0, 0.0, 1.0, 46014.02, 85047.44, 205517.64],
        [0.0, 0.0, 1.0, 114523.61, 122616.84, 261776.23],
        [0.0, 0.0, 1.0, 144372.41, 118671.85, 383199.62],
        [0.0, 0.0, 1.0, 1000.23, 124153.04, 1903.93],
        [0.0, 0.0, 1.0, 120542.52, 148718.95, 311613.29],
        [1.0, 0.0, 0.0, 123334.88, 108679.17, 304981.62],
        [0.0, 0.0, 1.0, 72107.6, 127864.55, 353183.81],
        [0.0, 0.0, 1.0, 542.05, 51743.15, 0.0],
        [1.0, 0.0, 0.0, 0.0, 116983.8, 45173.06],
        [0.0, 0.0, 1.0, 131876.9, 99814.71, 362861.36],
        [1.0, 0.0, 0.0, 76253.86, 113867.3, 298664.47],
        [0.0, 1.0, 0.0, 1315.46, 115816.21, 297114.46],
        [0.0, 0.0, 1.0, 165349.2, 136897.8, 471784.1],
        [0.0, 0.0, 1.0, 55493.95, 103057.49, 214634.81],
        [1.0, 0.0, 0.0, 27892.92, 84710.77, 164470.71],
        [1.0, 0.0, 0.0, 91992.39, 135495.07, 252664.93],
        [1.0, 0.0, 0.0, 142107.34, 91391.77, 366168.42],
        [1.0, 0.0, 0.0, 63408.86, 129219.61, 46085.25],
        [1.0, 0.0, 0.0, 67532.53, 105751.03, 304768.73],
        [1.0, 0.0, 0.0, 64664.71, 139553.16, 137962.62],
        [0.0, 0.0, 1.0, 75328.87, 144135.98, 134050.07],
        [1.0, 0.0, 0.0, 78013.11, 121597.55, 264346.06],
        [1.0, 0.0, 0.0, 101913.08, 110594.11, 229160.95],
        [0.0, 1.0, 0.0, 66051.52, 182645.56, 118148.2],
        [1.0, 0.0, 0.0, 23640.93, 96189.63, 148001.11],
        [0.0, 0.0, 1.0, 20229.59, 65947.93, 185265.1],
        [0.0, 0.0, 1.0, 130298.13, 145530.06, 323876.68],
```

```
       [1.0, 0.0, 0.0, 73994.56, 122782.75, 303319.26],
       [1.0, 0.0, 0.0, 93863.75, 127320.38, 249839.44],
       [0.0, 0.0, 1.0, 77044.01, 99281.34, 140574.81],
       [0.0, 0.0, 1.0, 15505.73, 127382.3, 35534.17]], dtype=object),
array([[1.0, 0.0, 0.0, 0.0, 135426.92, 0.0],
       [0.0, 0.0, 1.0, 78389.47, 153773.43, 299737.29],
       [0.0, 1.0, 0.0, 119943.24, 156547.42, 256512.92],
       [0.0, 0.0, 1.0, 61994.48, 115641.28, 91131.24],
       [1.0, 0.0, 0.0, 44069.95, 51283.14, 197029.42],
       [0.0, 1.0, 0.0, 28754.33, 118546.05, 172795.67],
       [0.0, 1.0, 0.0, 86419.7, 153514.11, 0.0],
       [1.0, 0.0, 0.0, 162597.7, 151377.59, 443898.53],
       [0.0, 1.0, 0.0, 153441.51, 101145.55, 407934.54],
       [1.0, 0.0, 0.0, 134615.46, 147198.87, 127716.82]], dtype=object),
array([ 90708.19,  65200.33, 125370.37, 144259.4 ,  81005.76, 101004.64,
        97483.56, 124266.9 ,  96712.8 ,  96479.51, 129917.04, 182901.99,
        64926.08, 152211.77, 149759.96, 105008.31,  35673.41,  14681.4 ,
       156991.12, 118474.03,  49490.75, 192261.83,  96778.92,  77798.83,
       134307.35, 166187.94,  97427.84, 108733.99, 107404.34, 105733.54,
       126992.93, 146121.95, 103282.38,  71498.49,  81229.06, 155752.6 ,
       110352.25, 141585.52, 108552.04,  69758.98]),
array([ 42559.73, 111313.02, 132602.65,  99937.59,  89949.14,  78239.91,
       122776.86, 191792.06, 191050.39, 156122.51])]
```

[126]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.
 ↪2,random_state=14)
```

[127]:
```
print(x.shape)
print(y.shape)
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(50, 6)
(50,)
(40, 6)
(10, 6)
(40,)
(10,)
```

[128]:
```
from sklearn.linear_model import LinearRegression
l=LinearRegression()
l=l.fit(x_train,y_train)
l
```

[128]:
```
LinearRegression()
```

[129]:
```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
Label=LabelEncoder()
x[:,3]=Label.fit_transform(x[:,3])
```

[130]:
```
print(x)
```

```
[[0.0 0.0 1.0 48 136897.8 471784.1]
 [1.0 0.0 0.0 47 151377.59 443898.53]
 [0.0 1.0 0.0 46 101145.55 407934.54]
 [0.0 0.0 1.0 45 118671.85 383199.62]
 [1.0 0.0 0.0 44 91391.77 366168.42]
 [0.0 0.0 1.0 42 99814.71 362861.36]
 [1.0 0.0 0.0 43 147198.87 127716.82]
 [0.0 0.0 1.0 41 145530.06 323876.68]
 [0.0 0.0 1.0 39 148718.95 311613.29]
 [1.0 0.0 0.0 40 108679.17 304981.62]
 [1.0 0.0 0.0 36 110594.11 229160.95]
 [0.0 1.0 0.0 35 91790.61 249744.55]
 [1.0 0.0 0.0 33 127320.38 249839.44]
 [1.0 0.0 0.0 32 135495.07 252664.93]
 [0.0 1.0 0.0 38 156547.42 256512.92]
 [0.0 0.0 1.0 37 122616.84 261776.23]
 [1.0 0.0 0.0 28 121597.55 264346.06]
 [0.0 0.0 1.0 34 145077.58 282574.31]
 [0.0 0.0 1.0 31 114175.79 294919.57]
 [0.0 1.0 0.0 30 153514.11 0.0]
 [1.0 0.0 0.0 26 113867.3 298664.47]
 [0.0 0.0 1.0 29 153773.43 299737.29]
 [1.0 0.0 0.0 24 122782.75 303319.26]
 [1.0 0.0 0.0 22 105751.03 304768.73]
 [0.0 0.0 1.0 27 99281.34 140574.81]
 [1.0 0.0 0.0 19 139553.16 137962.62]
 [0.0 0.0 1.0 25 144135.98 134050.07]
 [0.0 0.0 1.0 23 127864.55 353183.81]
 [0.0 1.0 0.0 21 182645.56 118148.2]
 [0.0 0.0 1.0 20 153032.06 107138.38]
 [0.0 0.0 1.0 17 115641.28 91131.24]
 [0.0 0.0 1.0 16 152701.92 88218.23]
 [1.0 0.0 0.0 18 129219.61 46085.25]
 [0.0 0.0 1.0 15 103057.49 214634.81]
 [0.0 1.0 0.0 14 157693.92 210797.67]
 [0.0 0.0 1.0 13 85047.44 205517.64]
 [0.0 0.0 1.0 9 127056.21 201126.82]
 [1.0 0.0 0.0 12 51283.14 197029.42]
 [0.0 0.0 1.0 5 65947.93 185265.1]
```

```
[1.0 0.0 0.0 11 82982.09 174999.3]
[0.0 1.0 0.0 10 118546.05 172795.67]
[1.0 0.0 0.0 8 84710.77 164470.71]
[1.0 0.0 0.0 7 96189.63 148001.11]
[0.0 0.0 1.0 4 127382.3 35534.17]
[1.0 0.0 0.0 6 154806.14 28334.72]
[0.0 0.0 1.0 2 124153.04 1903.93]
[0.0 1.0 0.0 3 115816.21 297114.46]
[1.0 0.0 0.0 0 135426.92 0.0]
[0.0 0.0 1.0 1 51743.15 0.0]
[1.0 0.0 0.0 0 116983.8 45173.06]]
```

[131]:
```
from sklearn.compose import ColumnTransformer
ct=ColumnTransformer([('encoder',OneHotEncoder(),[3])],remainder='passthrough')
x=np.array(ct.fit_transform(x))
print(x)
```

```
  (0, 48)       1.0
  (0, 51)       1.0
  (0, 52)       136897.8
  (0, 53)       471784.1
  (1, 47)       1.0
  (1, 49)       1.0
  (1, 52)       151377.59
  (1, 53)       443898.53
  (2, 46)       1.0
  (2, 50)       1.0
  (2, 52)       101145.55
  (2, 53)       407934.54
  (3, 45)       1.0
  (3, 51)       1.0
  (3, 52)       118671.85
  (3, 53)       383199.62
  (4, 44)       1.0
  (4, 49)       1.0
  (4, 52)       91391.77
  (4, 53)       366168.42
  (5, 42)       1.0
  (5, 51)       1.0
  (5, 52)       99814.71
  (5, 53)       362861.36
  (6, 43)       1.0
  :     :
  (43, 51)      1.0
  (43, 52)      127382.3
  (43, 53)      35534.17
  (44, 6)       1.0
  (44, 49)      1.0
```

```
(44, 52)      154806.14
(44, 53)      28334.72
(45, 2)       1.0
(45, 51)      1.0
(45, 52)      124153.04
(45, 53)      1903.93
(46, 3)       1.0
(46, 50)      1.0
(46, 52)      115816.21
(46, 53)      297114.46
(47, 0)       1.0
(47, 49)      1.0
(47, 52)      135426.92
(48, 1)       1.0
(48, 51)      1.0
(48, 52)      51743.15
(49, 0)       1.0
(49, 49)      1.0
(49, 52)      116983.8
(49, 53)      45173.06
```

[132]:
```
y_pred=l.predict(x_test)
y_pred
```

[132]:
```
array([ 46992.29340222, 118113.93670713, 152546.84491681,  99206.91723847,
        87945.53013534,  77273.68551212, 118522.64975353, 189567.93038269,
       183707.23325005, 158307.55876047])
```

[133]:
```
y_test
```

[133]:
```
array([ 42559.73, 111313.02, 132602.65,  99937.59,  89949.14,  78239.91,
       122776.86, 191792.06, 191050.39, 156122.51])
```

[134]:
```
result=pd.concat([pd.DataFrame(y_pred,columns=['Prediction']),
                  pd.DataFrame(y_test,columns=['Test'])],axis=1)
result['Difference']=result['Prediction']-result['Test']
result
```

[134]:

	Prediction	Test	Difference
0	46992.293402	42559.73	4432.563402
1	118113.936707	111313.02	6800.916707
2	152546.844917	132602.65	19944.194917
3	99206.917238	99937.59	-730.672762
4	87945.530135	89949.14	-2003.609865
5	77273.685512	78239.91	-966.224488
6	118522.649754	122776.86	-4254.210246
7	189567.930383	191792.06	-2224.129617

```
8  183707.233250  191050.39  -7343.156750
9  158307.558760  156122.51   2185.048760
```

[135]:
```
r2_score(y_test,y_pred)
```

[135]:
```
0.9734749543551718
```

[150]:
```
y_final_pred=l.predict(x)
y_final_pred
```

[150]:
```
array([192577.59361311, 189567.93038269, 183707.23325005, 173334.25307143,
       171093.01766635, 162795.31959542, 158307.55876047, 160356.3667628 ,
       152197.73705585, 154319.25773265, 135041.77592031, 137046.83963991,
       129148.83111479, 127715.54946377, 152546.84491681, 146028.4935784 ,
       116878.25108624, 130669.58422671, 128743.35102725, 118522.64975353,
       116446.53136456, 118113.93670713, 114752.71224431, 109651.22374274,
       112669.56876675, 102616.16692774, 111033.53776076, 114631.09644782,
       105481.00894923, 102479.60216101,  99206.91723847,  98372.14228427,
        99052.0934655 ,  97490.44026376,  92415.21124119,  89678.59702283,
        75592.29185814,  87945.53013534,  68505.31333551,  82859.28080479,
        77273.68551212,  74023.52399788,  70137.50109256,  60414.2085062 ,
        65504.7985268 ,  47865.59922452,  58802.27016968,  46992.29340222,
        47574.76343747,  48292.39484897])
```

[151]:
```
y_fpred=pd.DataFrame(y_final_pred,columns=['Prediction'])
result=pd.concat([df2,y_fpred],axis=1)
result['Difference']=result['Profit']-result['Prediction']
result
```

[151]:

	R&D Spend	Administration	Marketing Spend	State	Profit	\
0	165349.20	136897.80	471784.10	New York	192261.83	
1	162597.70	151377.59	443898.53	California	191792.06	
2	153441.51	101145.55	407934.54	Florida	191050.39	
3	144372.41	118671.85	383199.62	New York	182901.99	
4	142107.34	91391.77	366168.42	California	166187.94	
5	131876.90	99814.71	362861.36	New York	156991.12	
6	134615.46	147198.87	127716.82	California	156122.51	
7	130298.13	145530.06	323876.68	New York	155752.60	
8	120542.52	148718.95	311613.29	New York	152211.77	
9	123334.88	108679.17	304981.62	California	149759.96	
10	101913.08	110594.11	229160.95	California	146121.95	
11	100671.96	91790.61	249744.55	Florida	144259.40	
12	93863.75	127320.38	249839.44	California	141585.52	
13	91992.39	135495.07	252664.93	California	134307.35	
14	119943.24	156547.42	256512.92	Florida	132602.65	
15	114523.61	122616.84	261776.23	New York	129917.04	
16	78013.11	121597.55	264346.06	California	126992.93	

17	94657.16	145077.58	282574.31	New York	125370.37
18	91749.16	114175.79	294919.57	New York	124266.90
19	86419.70	153514.11	0.00	Florida	122776.86
20	76253.86	113867.30	298664.47	California	118474.03
21	78389.47	153773.43	299737.29	New York	111313.02
22	73994.56	122782.75	303319.26	California	110352.25
23	67532.53	105751.03	304768.73	California	108733.99
24	77044.01	99281.34	140574.81	New York	108552.04
25	64664.71	139553.16	137962.62	California	107404.34
26	75328.87	144135.98	134050.07	New York	105733.54
27	72107.60	127864.55	353183.81	New York	105008.31
28	66051.52	182645.56	118148.20	Florida	103282.38
29	65605.48	153032.06	107138.38	New York	101004.64
30	61994.48	115641.28	91131.24	New York	99937.59
31	61136.38	152701.92	88218.23	New York	97483.56
32	63408.86	129219.61	46085.25	California	97427.84
33	55493.95	103057.49	214634.81	New York	96778.92
34	46426.07	157693.92	210797.67	Florida	96712.80
35	46014.02	85047.44	205517.64	New York	96479.51
36	28663.76	127056.21	201126.82	New York	90708.19
37	44069.95	51283.14	197029.42	California	89949.14
38	20229.59	65947.93	185265.10	New York	81229.06
39	38558.51	82982.09	174999.30	California	81005.76
40	28754.33	118546.05	172795.67	Florida	78239.91
41	27892.92	84710.77	164470.71	California	77798.83
42	23640.93	96189.63	148001.11	California	71498.49
43	15505.73	127382.30	35534.17	New York	69758.98
44	22177.74	154806.14	28334.72	California	65200.33
45	1000.23	124153.04	1903.93	New York	64926.08
46	1315.46	115816.21	297114.46	Florida	49490.75
47	0.00	135426.92	0.00	California	42559.73
48	542.05	51743.15	0.00	New York	35673.41
49	0.00	116983.80	45173.06	California	14681.40

	Prediction	Difference
0	192577.593613	-315.763613
1	189567.930383	2224.129617
2	183707.233250	7343.156750
3	173334.253071	9567.736929
4	171093.017666	-4905.077666
5	162795.319595	-5804.199595
6	158307.558760	-2185.048760
7	160356.366763	-4603.766763
8	152197.737056	14.032944
9	154319.257733	-4559.297733
10	135041.775920	11080.174080
11	137046.839640	7212.560360

```
12  129148.831115  12436.688885
13  127715.549464   6591.800536
14  152546.844917 -19944.194917
15  146028.493578 -16111.453578
16  116878.251086  10114.678914
17  130669.584227  -5299.214227
18  128743.351027  -4476.451027
19  118522.649754   4254.210246
20  116446.531365   2027.498635
21  118113.936707  -6800.916707
22  114752.712244  -4400.462244
23  109651.223743   -917.233743
24  112669.568767  -4117.528767
25  102616.166928   4788.173072
26  111033.537761  -5299.997761
27  114631.096448  -9622.786448
28  105481.008949  -2198.628949
29  102479.602161  -1474.962161
30   99206.917238    730.672762
31   98372.142284   -888.582284
32   99052.093465  -1624.253465
33   97490.440264   -711.520264
34   92415.211241   4297.588759
35   89678.597023   6800.912977
36   75592.291858  15115.898142
37   87945.530135   2003.609865
38   68505.313336  12723.746664
39   82859.280805  -1853.520805
40   77273.685512    966.224488
41   74023.523998   3775.306002
42   70137.501093   1360.988907
43   60414.208506   9344.771494
44   65504.798527   -304.468527
45   47865.599225  17060.480775
46   58802.270170  -9311.520170
47   46992.293402  -4432.563402
48   47574.763437 -11901.353437
49   48292.394849 -33610.994849
```

[152]:
```
r2_score(df2['Profit'],y_final_pred)
```

[152]:
```
0.9506036891520072
```

[139]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
import warnings
warnings.filterwarnings('ignore')
from sklearn.metrics import r2_score,mean_squared_error,mean_absolute_error
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

[65]:
```
df=pd.read_csv('Position Salary .csv')
df.head()
```

[65]:

	Position	Level	Salary
0	Busienss Analyst	1	45000
1	Junior Consultant	2	50000
2	Senior Consultant	3	60000
3	Manager	4	120000
4	Country Manager	5	140000

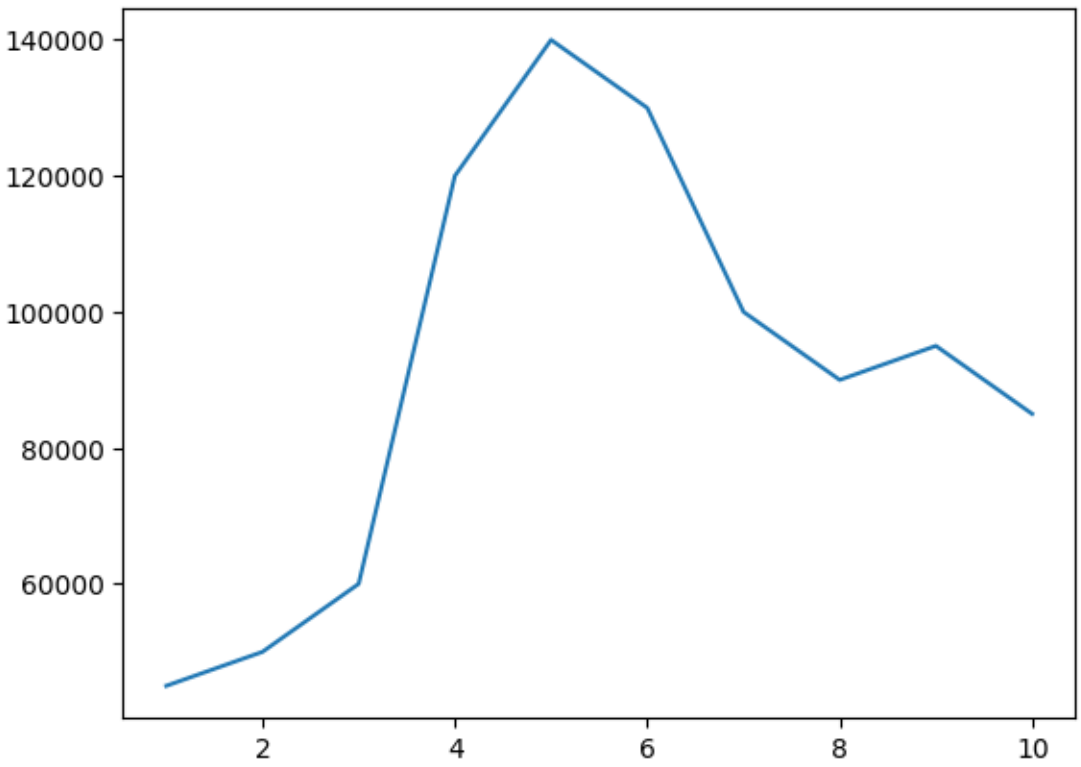
[66]:
```
x=df.iloc[:,1:2].values
print(x)
```

```
[[ 1]
 [ 2]
 [ 3]
 [ 4]
 [ 5]
 [ 6]
 [ 7]
 [ 8]
 [ 9]
 [10]]
```

[67]:
```
y=df.iloc[:,2].values
print(y)
```

```
[ 45000  50000  60000 120000 140000 130000 100000  90000  95000  85000]
```

[68]:
```
plt.plot(x,y)
plt.show()
```



[69]:
```
from sklearn.tree import DecisionTreeRegressor
dt=DecisionTreeRegressor()
dt.fit(x,y)
print(dt)
```

```
DecisionTreeRegressor()
```

[70]:
```
y_pred=dt.predict(x)
print(y_pred)
```

```
[ 45000.  50000.  60000. 120000. 140000. 130000. 100000.  90000.  95000.
  85000.]
```

[71]:
```
r2_score(y,y_pred)
```

[71]: 1.0

[72]:
```
from sklearn.linear_model import LinearRegression
lr=LinearRegression()
lr.fit(x,y)
print(lr)
```

```
LinearRegression()
```

[73]:
```
y_pred=lr.predict(x)
```

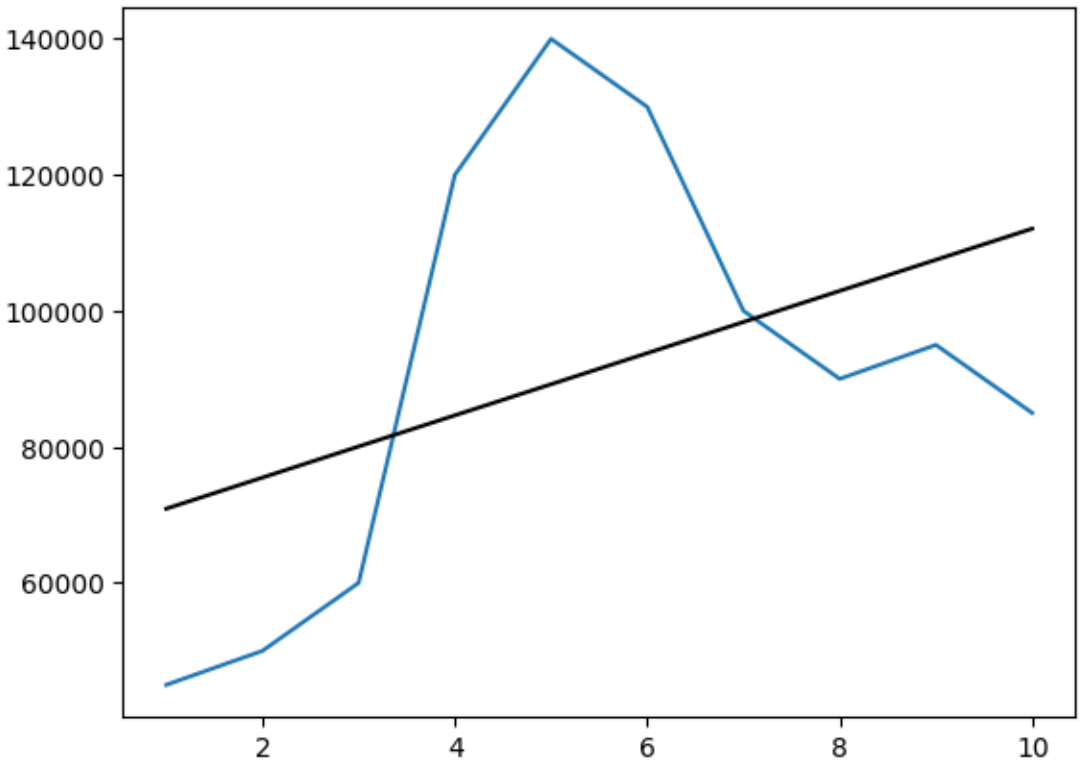
[74]:
```
y_pred
```

[74]:
```
array([ 70909.09090909,  75484.84848485,  80060.60606061,  84636.36363636,
        89212.12121212,  93787.87878788,  98363.63636364, 102939.39393939,
       107515.15151515, 112090.90909091])
```

[75]:
```
r2_score(y,y_pred)
```

[75]:
```
0.17895348198378502
```

[76]:
```
plt.plot(x,y)
plt.plot(x,y_pred,color='black')
plt.show()
```



[ ]:

[ ]: