In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import statsmodels.formula.api as smf
from sklearn.metrics import mean_absolute_error,mean_squared_error,r2_score

import warnings
warnings.filterwarnings("ignore")
```

In [4]:
```
dataset
```

Out[4]:

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

	DigitalMarketing	Promotion	Research	State	Profit
33	55493.95	103057.49	214634.81	Chennai	96778.92
34	46426.07	157693.92	210797.67	Bangalore	96712.80
35	46014.02	85047.44	205517.64	Hyderabad	96479.51
36	28663.76	127056.21	201126.82	Chennai	90708.19
37	44069.95	51283.14	197029.42	Bangalore	89949.14
38	20229.59	65947.93	185265.10	Hyderabad	81229.06
39	38558.51	82982.09	174999.30	Bangalore	81005.76
40	28754.33	118546.05	172795.67	Bangalore	78239.91
41	27892.92	84710.77	164470.71	Chennai	77798.83
42	23640.93	96189.63	148001.11	Bangalore	71498.49
43	15505.73	127382.30	35534.17	Hyderabad	69758.98
44	22177.74	154806.14	28334.72	Bangalore	65200.33
45	1000.23	124153.04	1903.93	Hyderabad	64926.08
46	1315.46	115816.21	297114.46	Chennai	49490.75
47	0.00	135426.92	0.00	Bangalore	42559.73
48	542.05	51743.15	0.00	Hyderabad	35673.41
49	0.00	116983.80	45173.06	Bangalore	14681.40

In [5]:
```
X = dataset.iloc[:, :-1]
y = dataset.iloc[:, 4]
X=pd.get_dummies(X,dtype=int)
```

In [6]:
```
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_sta
```

In [7]:
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
```

In [8]:
```
regressor.fit(X_train, y_train)
y_pred = regressor.predict(X_test)
```

In [9]:
```
print("r2:",r2_score(y_pred,y_test))
```

```
r2: 0.9293749209318107
```

In [10]:
```
dataset.columns
```

Out[10]:
```
Index(['DigitalMarketing', 'Promotion', 'Research', 'State', 'Profit'], dtype
='object')
```

In [11]:
```
model = smf.ols('Profit ~ DigitalMarketing + Promotion + Research + State', data
```

In [12]:
```
model.summary()
```

Out[12]:

OLS Regression Results

Dep. Variable:	Profit	**R-squared:**	0.951
Model:	OLS	**Adj. R-squared:**	0.945
Method:	Least Squares	**F-statistic:**	169.9
Date:	Wed, 06 Nov 2024	**Prob (F-statistic):**	1.34e-27
Time:	17:39:35	**Log-Likelihood:**	-525.38
No. Observations:	50	**AIC:**	1063.
Df Residuals:	44	**BIC:**	1074.
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	5.013e+04	6884.820	7.281	0.000	3.62e+04	6.4e+04
State[T.Chennai]	198.7888	3371.007	0.059	0.953	-6595.030	6992.607
State[T.Hyderabad]	-41.8870	3256.039	-0.013	0.990	-6604.003	6520.229
DigitalMarketing	0.8060	0.046	17.369	0.000	0.712	0.900
Promotion	-0.0270	0.052	-0.517	0.608	-0.132	0.078
Research	0.0270	0.017	1.574	0.123	-0.008	0.062

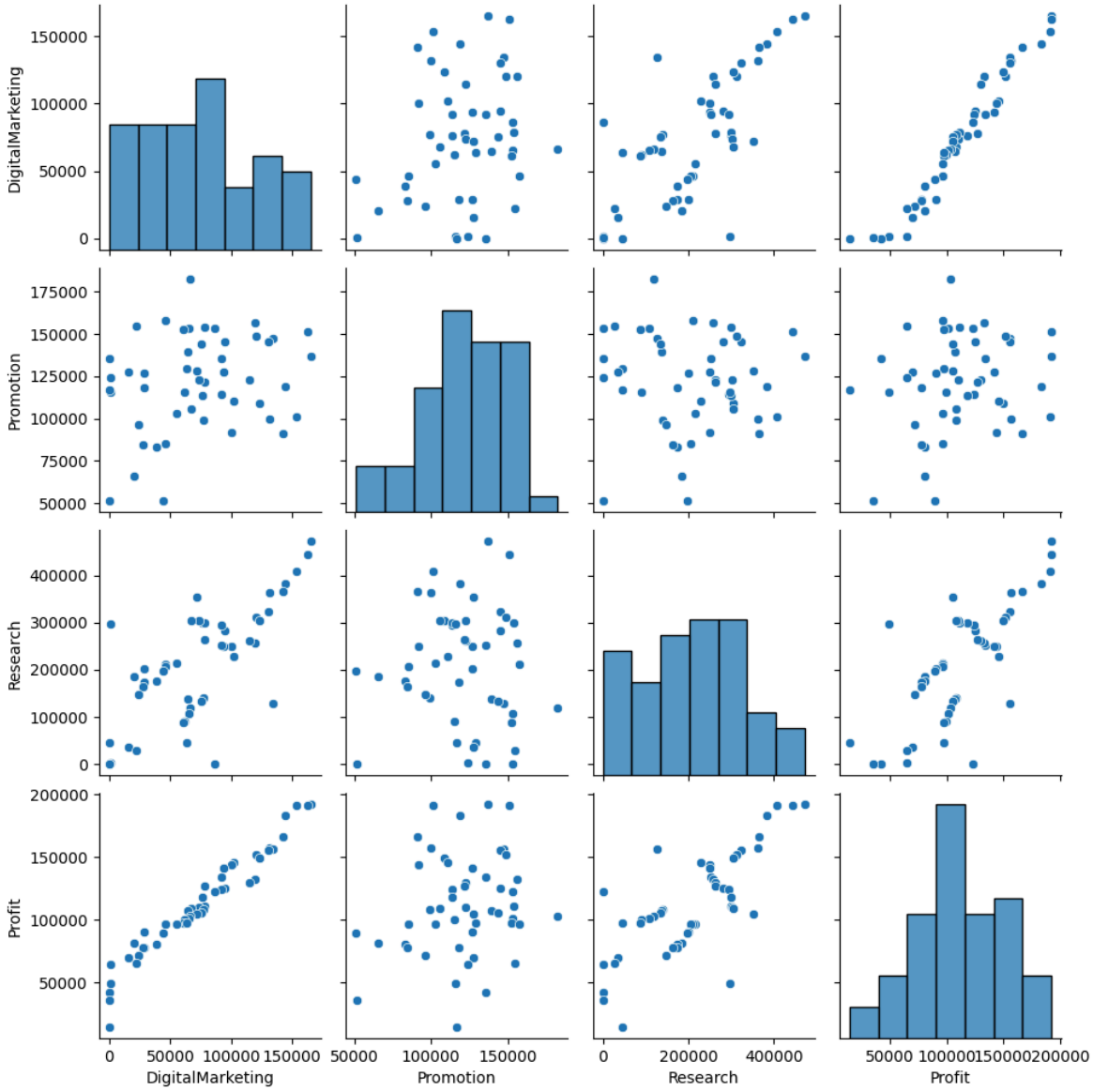
Omnibus:	14.782	**Durbin-Watson:**	1.283
Prob(Omnibus):	0.001	**Jarque-Bera (JB):**	21.266
Skew:	-0.948	**Prob(JB):**	2.41e-05
Kurtosis:	5.572	**Cond. No.**	1.45e+06

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.45e+06. This might indicate that there are strong multicollinearity or other numerical problems.

In [13]:
```
import seaborn as sns

sns.pairplot(dataset)
```

Out[13]:
```
<seaborn.axisgrid.PairGrid at 0x1bba20400d0>
```



In [16]:
```
finalmodal=smf.ols('Profit~Promotion',data=dataset).fit()
finalmodal.rsquared
```

Out[16]: 0.04028714077757223

In [17]:
```
finalmodal=smf.ols('Profit~Research',data=dataset).fit()
finalmodal.rsquared
```

Out[17]: 0.5591535746115515

In [14]:
```
finalmodal=smf.ols('Profit~DigitalMarketing',data=dataset).fit()
```

In [15]:
```
finalmodal.rsquared
```

Out[15]: 0.9465353160804392

In [ ]: