ASSIGNMENT/ TASK 8

GO_STP_12574
SHWETA JHA

Task- Predicting a Startups Profit/Success Rate using Multiple Linear Regression in Python- Download Data Set click here. Here 50 startups dataset containing 5 columns like “R&D Spend”, “Administration”, “Marketing Spend”, “State”, “Profit”.

In this dataset first 3 columns provides you spending on Research , Administration and Marketing respectively. State indicates startup based on that state. Profit indicates how much profits earned by a startup.

Clearly, we can understand that it is a multiple linear regression problem, as the independent variables are more than one.

Prepare a prediction model for profit of 50_Startups data in Python

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.linear_model import LinearRegression
```

```
dataset=pd.read_csv('/content/50_Startups.csv')
dataset.head()
```

	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	Florida	166187.94

```
dataset.info()
dataset.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   R&D Spend        50 non-null     float64
 1   Administration   50 non-null     float64
 2   Marketing Spend  50 non-null     float64
 3   State            50 non-null     object 
 4   Profit           50 non-null     float64
dtypes: float64(4), object(1)
memory usage: 2.1+ KB
```

	R&D Spend	Administration	Marketing Spend	Profit
count	50.000000	50.000000	50.000000	50.000000
mean	73721.615600	121344.639600	211025.097800	112012.639200
std	45902.256482	28017.802755	122290.310726	40306.180338
min	0.000000	51283.140000	0.000000	14681.400000
25%	39936.370000	103730.875000	129300.132500	90138.902500
50%	73051.080000	122699.795000	212716.240000	107978.190000
75%	101602.800000	144842.180000	299469.085000	139765.977500

```
dataset.corr()
```

	R&D Spend	Administration	Marketing Spend	Profit
R&D Spend	1.000000	0.241955	0.724248	0.972900
Administration	0.241955	1.000000	-0.032154	0.200717
Marketing Spend	0.724248	-0.032154	1.000000	0.747766
Profit	0.972900	0.200717	0.747766	1.000000

```
X = dataset.iloc[:, :-1]
y = dataset.iloc[:, 4]


states=pd.get_dummies(X['State'],drop_first=True)
states.head()
```

	Florida	New York
0	0	1
1	0	0
3	0	1

```
X=X.drop('State',axis=1)
X=pd.concat([X,states],axis=1)
```

```
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 = 0)
print(X_train)
print(X_test)
print(y_train)
print(y_test)
print(X_train.shape)
print(X_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
      38   20229.59        65947.93        185265.10        0         1
      31   61136.38       152701.92         88218.23        0         1
      22   73994.56       122782.75        303319.26        1         0
      4   142107.34        91391.77        366168.42        1         0
      33     96778.92
      35     96479.51
      26    105733.54
      34     96712.80
      18    124266.90
      7     155752.60
      14    132602.65
      45     64926.08
      48     35673.41
      29    101004.64
      15    129917.04
      30     99937.59
      32     97427.84
      16    126992.93
      42     71498.49
      20    118474.03
      43     69758.98
      8     152211.77
      13    134307.35
      25    107404.34
      5     156991.12
      17    125370.37
      40     78239.91
      49     14681.40
      1     191792.06
      12    141585.52
      37     89949.14
      24    108552.04
      6     156122.51
      23    108733.99
      36     90708.19
      21    111313.02
```

```
19    122776.86
9     149759.96
39     81005.76
46     49490.75
3     182901.99
0     192261.83
47     42559.73
44     65200.33

Name: Profit, dtype: float64
28    103282.38
11    144259.40
10    146121.95
41     77798.83
2     191050.39
27    105008.31
38     81229.06
31     97483.56
22    110352.25
4     166187.94
Name: Profit, dtype: float64
(40, 5)
(10, 5)
(40,)
(10,)
```

```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor
```

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

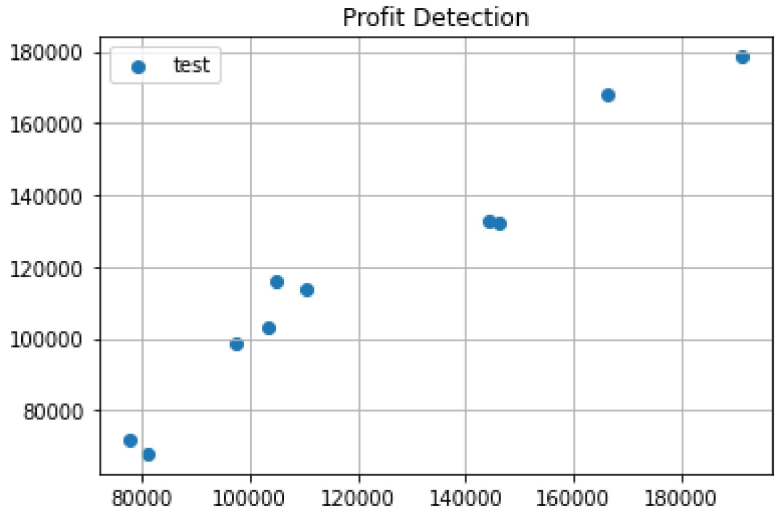
```
regressor.fit(X_train, y_train)
```

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

```
y_pred = regressor.predict(X_test)
from sklearn.metrics import r2_score
score=r2_score(y_test,y_pred)
score
```

```
0.9347068473282423
```

```
plt.scatter(y_test, y_pred,label='test')
plt.title('Profit Detection')
plt.legend()
plt.grid()
plt.show()
```



✓ 0s completed at 2:13 PM