1.Multiple linear regression

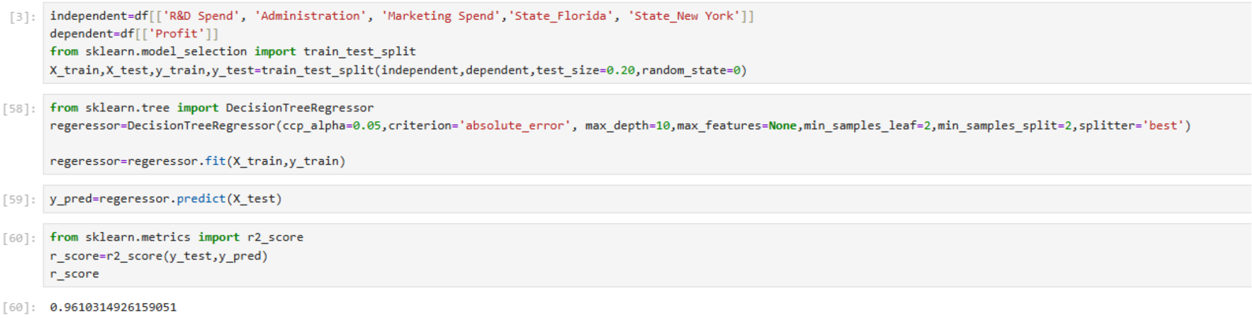
R2 score: 0.9347068473282423

2.Support Vector Machine

S. No	Hyperperameter Tuning	***linear***	***poly***	***rbf***	***sigmoid***	***precomputed***
	0.01	0.9395297184086321	-0.159757126895115	-0.1598747132296754	-0.15987903150761107	Error
	0.1	0.9335952989928425	-0.15983825946559072	-0.15983825946559072	-0.15988144165794327	
	1	0.874257419393536	-0.1479333450191409	-0.1594737803854509	-0.15990554360783116	
	10	-0.1336914357017669	-0.1336914357017669	-0.1336914357017669		
	100	-286.1920043916585	-0.1336914357017669	-0.1336914357017669		
	1000		-0.1336914357017669	-0.1336914357017669	0.133691435701766	

3.DecisionTree

Best combination of hyperparameters for a DecisionTreeRegressor



```
[3]:  independent=df[['R&D Spend', 'Administration', 'Marketing Spend','State_Florida', 'State_New York']]
      dependent=df[['Profit']]
      from sklearn.model_selection import train_test_split
      X_train,X_test,y_train,y_test=train_test_split(independent,dependent,test_size=0.20,random_state=0)

[58]: from sklearn.tree import DecisionTreeRegressor
      regeressor=DecisionTreeRegressor(ccp_alpha=0.05,criterion='absolute_error', max_depth=10,max_features=None,min_samples_leaf=2,min_samples_split=2,splitter='best')

      regeressor=regeressor.fit(X_train,y_train)

[59]: y_pred=regeressor.predict(X_test)

[60]: from sklearn.metrics import r2_score
      r_score=r2_score(y_test,y_pred)
      r_score

[60]: 0.9610314926159051
```

S.NO	Criterion	Ssplitter	Max_features	R² score
1	absolute_error	best	sqrt	0.9461114759249466
2	absolute_error	random	sqrt	0.7507202700961755
3	absolute_error	best	log2	0.17833136031032748
4	absolute_error	random	log2	0.4351864063010229
5	*squared_error*	best	sqrt	0.8668488337805973
6	*squared_error*	random	sqrt	0.8142426573631658
7	*squared_error*	best	log2	0.5903662729507224
8	*squared_error*	random	log2	0.5896235132407462
9	*friedman_mse*	best	sqrt	0.6047709061434949
10	*friedman_mse*	random	sqrt	0.8035256166244932
11	*friedman_mse*	best	log2	0.04643686875627029
12	*friedman_mse*	random	log2	-0.29332798702686014
13	*poisson*	best	sqrt	0.6191721066916681
14	*poisson*	random	sqrt	0.8009987801521994
15	*poisson*	best	log2	0.6191721066916681
16.	*poisson*	random	log2	0.8583370280613273