R^2 Values for Various Algorithm in ML

Data – 50_Startups

Goal – To predict Profit

1. Multiple Linear Regression

R^2 = 0.9358680970046241

2. Support Vector Machine

Model#	kernel	C	R^2
1	poly	1	-0.050896586
2	poly	10	0.025319733
3	poly	100	0.465666056
4	poly	1000	0.640328112
5	rbf	1	-0.057323757
6	rbf	10	-0.055807402
7	rbf	100	-0.030227628
8	rbf	1000	0.16060299
9	sigmoid	1	-0.057505641
10	sigmoid	10	-0.057621827
11	sigmoid	100	-0.058786436
12	sigmoid	1000	-0.070707399

So, in **Support Vector Machine Algorithm** best model came for

*Kernel = **poly***
C = 1000

R^2 = 0.640328112

3. Decision Tree

Model#	criterion	splitter	max_features	R^2
1	squared_error	best	sqrt	0.908781285
2	squared_error	best	log2	0.499265724
3	squared_error	random	sqrt	0.441262221
4	squared_error	random	log2	0.3463428
5	friedman_mse	best	sqrt	0.360082248
6	friedman_mse	best	log2	0.936117097
7	friedman_mse	random	sqrt	0.681760422
8	friedman_mse	random	log2	0.504279522
9	absolute_error	best	sqrt	0.637962965
10	absolute_error	best	log2	0.47203096
11	absolute_error	random	sqrt	0.792476623
12	absolute_error	random	log2	-0.03308019
13	poisson	best	sqrt	0.522309807
14	poisson	best	log2	0.680850104
15	poisson	random	sqrt	-0.09132119
16	poisson	random	log2	0.713641648

So, in **Decision Tree Algorithm** best model came for

Criterion = **friedman_mse**
Splitter = **best**
max_features = **log2**

R^2 = 0.936117097

4. Random Forest

Model#	n_estimators	criterion	max_features	R2
1	50	squared_error	sqrt	0.684518937
2	50	squared_error	log2	0.684518937
3	50	absolute_error	sqrt	0.722177208
4	50	absolute_error	log2	0.722177208
5	50	friedman_mse	sqrt	0.684518937
6	50	friedman_mse	log2	0.684518937
7	50	poisson	sqrt	0.724294796
8	50	poisson	log2	0.724294796
9	75	squared_error	sqrt	0.695483524
10	75	squared_error	log2	0.695483524
11	75	absolute_error	sqrt	0.757959419
12	75	absolute_error	log2	0.757959419
13	75	friedman_mse	sqrt	0.695483524
14	75	friedman_mse	log2	0.695483524
15	75	poisson	sqrt	0.755621408
16	75	poisson	log2	0.755621408
17	100	squared_error	sqrt	0.759211288
18	100	squared_error	log2	0.759211288
19	100	absolute_error	sqrt	0.786927265
20	100	absolute_error	log2	0.786927265
21	100	friedman_mse	sqrt	0.759211288
22	100	friedman_mse	log2	0.759211288
23	100	poisson	sqrt	0.773212971
24	100	poisson	log2	0.773212971

So, in **Random Forest Algorithm** best model came for

*Criterion = **absolute_error***
n_estimatore = 100
*max_features = **sqrt***

$R^2 = 0.786927265$