<u>HOPE AI ASSIGNMENT - REGRESSION</u>
<u>Tabulation of R-Square values</u>

GitHub Link for dataset:
https://github.com/JayachandraPrabha/Assignment-Regression/blob/main/insurance_pre.csv

Problem Statement / Requirement:

A client's requirement is, he wants to predict the insurance charges based on several parameters. The Client has provided the dataset of the same.
As a data scientist, you must develop a model which will predict the insurance charges.

1.) Identify your problem statement
2.) Tell basic info about the dataset (Total number of rows, columns)
3.) Mention the pre-processing method if you're doing any (like converting string to number–nominal data)
4.) Develop a good model with r2_score.You can use any machine learning algorithm, You can create many models. Finally, you have to come up with the final model.
5.) All the research values (r2_score of the models) should be documented. (You can make a tabulation or screenshot of the results.)
6.) Mention your final model, justify why you have chosen the same.

Tabulation of all the research values (r2_score of the models):

1. Multiple Linear Regression(MLR) → R-Square value: 0.7978644236809904

2. Support Vector Machine (SVM) → R-Square value: 0.8631863171770662

S.No	C	kernel			
	Hyper Parameter	Linear	Radial Bias function (RBF) - Non linear	Poly	Sigmoid
1	0.1	-0.0960	-0.1052	-0.1045	-0.104
2	1.0	-0.0120	-0.0987	-0.0899	-0.0894
3	10	0.5007	-0.0423	0.0459	0.0427
4	100	0.6423	0.3547	0.6591	0.5353
5	1000	0.7501	0.8283	0.8631	0.1720

3. Decision Tree (DT) → R-Square value: 0.7971644236809041

S.No	criterion	max_features	splitter	R-Square value
1	squared_error	none	best	0.7212
2			random	0.7634

3		auto	best	0.7187
4			random	0.7029
5		sqrt	best	0.7516
6			random	0.7297
7		log2	best	0.7116
8			random	0.7267
9	friedman_mse	none	best	0.7426
10			random	0.7402
11		auto	best	0.7472
12			random	0.7971
13		sqrt	best	0.7146
14			random	0.7903
15		log2	best	0.6892
16			random	0.6915
17	absolute_error	none	best	0.7036
18			random	0.7220
19		auto	best	0.6711
20			random	0.7292
21		sqrt	best	0.7640
22			random	0.7880
23		log2	best	0.7127
24			random	0.6779
25	poisson	none	best	0.7679
26			random	0.7368
27		auto	best	0.7609
28			random	0.7386
29		sqrt	best	0.6102
30			random	0.7447
31		log2	best	0.7328
32			random	0.6865

4. Random Forest (RF) → R-Square value: 0.8953409033050059

S.No	criterion	max_features	n_estimators	R-Square value
1	squared_error	none	10	0.8685
2			100	0.8784
3		auto	10	0.8530
4			100	0.8670
5		sqrt	10	0.8838
6			100	0.8931
7		log2	10	0.8652
8			100	0.8953
9	friedman_mse	none	10	0.8743
10			100	0.8478
11		auto	10	0.8289
12			100	0.8735
13		sqrt	10	0.8920
14			100	0.8887
15		log2	10	0.8850
16			100	0.8933
17	absolute_error	none	10	0.8521
18			100	0.8730
19		auto	10	0.8636
20			100	0.8681
21		sqrt	10	0.8557
22			100	0.8932
23		log2	10	0.8930
24			100	0.8944
25	poisson	none	10	0.8507
26			100	0.8762
27		auto	10	0.8759
28			100	0.8745
29		sqrt	10	0.8762
30			100	0.8948

31		log2	10	0.8768
32			100	0.8923

Conclusion:

The finalized model is **Random Forest.**

After analyzing with various algorithms & with its hyper/tuning parameters whose r-square values (r2_score) were as follows:

S.No	Algorithm	R-square value(r2_score)	Model output
1	Multiple Linear Regression(MLR)	0.7978644236809904	Poor
2	Support Vector Machine (SVM)	0.8631863171770662	Moderate
3	Decision Tree (DT)	0.7971644236809041	Poor
4	Random Forest (RF)	0.8953409033050059	Good

Conclusion:

In conclusion, the best model obtained is **Random Forest**.

The random forest algorithm provided the r-square value (r2_score*) is 0.8953 (nearly 90% of the accuracy). Hence Random Forest algorithm is finalized as the best model.

*(r2_score i.e., The obtained r-square value nearing 0 → Poor model & the value 1 nearing 1→ Good model)