Regression Assignment

Problem Statement or Requirement:

A client's requirement is, he wants to predict the insurance charges based on the 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

Domain Selection: - Machine Learning

Learning Selection: - Supervised Learning

Regression

2.) Tell basic info about the dataset (Total number of rows, columns)

Total number of Rows: -1338

Columns: - 6

3.) Mention the pre-processing method if you're doing any (like converting string to number – nominal data)

Converting string to number: - One Hot Encoding Method

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 final model.

5.) All the research values (r2_score of the models) should be documented. (You can make tabulation or screenshot of the results.)

- **Multiple Linear Regression.**

  r^2 Value = 0.7894

- **Support Vector Machine (SVM)**

SN	Parameters	Linear r^2 Value	Poly r^2 Value	Rbf r^2 Value	Sigmoid r^2 Value
1	C=0.01	-0.0797	-0.0893	-0.0896	-0.0897
2	C=0.1	-0.1220	-0.0862	-0.0897	-0.0897
3	C=1	-0.1116	-0.0642	-0.0884	-0.8994
4	C=10	-0.0016	-0.0931	-0.0819	-0.0907
5	C=100000	0.7436	0.7638	0.5340	-11215.8556

- **Decision Tree**

r^2 Value = 0.7042

SN	WITH HYPER TUNING PARAMETERS	r^2 Value
1	***splitter****='best'*	0.6802
2	***criterion***='friedman_mse'	0.7179
3	***criterion***='absolute_error'	0.6613
4	***splitter***='random'	0.7101
5	***criterion***='poisson'	0.7251
6	***criterion=***'absolute_error'***,*** ***splitter=***'best'	0.6621

- **Random Forest**

r^2 Value = 0.8496

SN	WITH HYPER TUNING PARAMETERS	r^2 Value
1	n_estimators=50,random_state=0	0.8496
2	***criterion****='squared_error'*	0.8566
3	criterion='absolute_error'	0.8546
4	criterion='poisson'	0.8547
5	min_samples_split=2	0.8574
6	warm_start=False	0.8537

6.) Mention your final model, justify why u have chosen the same.

- **Random Forest**

(min_samples_split=2) $\mathbf{r^2}$ Value = 0.8574

I evaluate all models with the support of the $\mathbf{r^2}$ value, if the $\mathbf{r^2}$ value comes to almost 1, that model should be the best model, according to which the Random Forest model comes as the best model.