Hope Artificial Intelligence Assignment-Regression

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.

1.Identifying the problem statement:

It's purely based on numbers so, the given problem statement comes under

Stage-1 : **Machine Learning.**

Stage-2: Requirement is clear, Input and Output is present so in stage-2 it comes under **supervised learning.**

Stage-3: It's a numerical data which it shows that it comes under **Regression.**

2.Information about the dataset:

1338 rows × 6 columns

['age','bmi', 'children', 'charges', 'sex_male', 'smoker_yes']

Independent Dataset: Input

'age','bmi', 'children', 'sex_male', 'smoker_yes'

Dependent Dataset: Output

'charges'

3.Data Preprocessing Method:

Columns : **Sex and Smoker** is given in strings in order to convert to numbers(**Nominal Data**) we have to change the dataset to binary so that we are getting the dummies from pandas libraries.

To find the Machine Learning Regression method using the R_Score value

1.Multiple Linear Regression : R_Score Value = 0.7894. accuracy

2.Support Vector Machine:

S.No	Hyper Parameter	Linear (R-Score)	Rbf (Non Linear) R-Score	Poly R-Score	Sigmoid R-Score
1.	C=10	0.462	-0.032	0.387	0.039
2.	C=100	0.628	0.320	0.617	0.527

3.	C=500	0.763	0.664	0.826	0.444
4.	C=1000	0.764	0.810	0.856	0.287
5.	C=5000	0.741	0.874	0.859	-7.53
6.	C=10000	0.741	0.877	0.859	-34.15

SVM Regression for R_Score Value : 0.877 accuracy Rbf (Non Linear) and hyper parameter C=10000.

3.Decion Tree:

S.No	Criterion	Splitter	Features	R-Score
1.	*squared_error*	*Best*	*Auto*	0.687
2.	*squared_error*	*Best*	*Sqrt*	0.621
3.	*squared_error*	*Best*	*log2*	0.740
4.	*squared_error*	*Random*	*Auto*	0.686
5.	*squared_error*	*Random*	*Sqrt*	0.670
6.	*squared_error*	*Random*	*log2*	0.618
7.	*friedman_mse*	*Best*	*Auto*	0.690
8.	*friedman_mse*	*Best*	*Sqrt*	0.646
9.	*friedman_mse*	*Best*	*log2*	0.730
10.	*friedman_mse*	*Random*	*Auto*	0.726
11.	*friedman_mse*	*Random*	*Sqrt*	0.672
12.	*friedman_mse*	*Random*	*log2*	0.718
13.	*mse*	*Best*	*Auto*	0.705
14.	*mse*	*Best*	*Sqrt*	0.741
15.	*mse*	*Best*	*log2*	0.740
16.	*mse*	*Random*	*Auto*	0.734
17.	*mse*	*Random*	*Sqrt*	0.700
18.	*mse*	*Random*	*log2*	0.626
19.	*mae*	*Best*	*Auto*	0.674
20.	*mae*	*Best*	*Sqrt*	0.713
21.	*mae*	*Best*	*log2*	0.704
22.	*mae*	*Random*	*Auto*	0.774
23.	*mae*	*Random*	*Sqrt*	0.641
24.	*mae*	*Random*	*log2*	0.720

The Decision Tree Regression R_Score Value for(mae,random,auto) is 0.774 accuracy

5.Random Forest:

S.No	Criterion	Max_features	N_estimators	R_score
1.	*squared_error*	*Auto*	10	0.833
2.	*squared_error*	*Sqrt*	10	0.852
3.	*squared_error*	*log2*	10	0.852
4.	*squared_error*	*Auto*	100	0.853
5.	*squared_error*	*Sqrt*	100	0.870
6.	*squared_error*	*log2*	100	0.870
7.	*friedman_mse*	*Auto*	10	0.833
8.	*friedman_mse*	*Sqrt*	10	0.850
9.	*friedman_mse*	*log2*	10	0.850
10.	*friedman_mse*	*Auto*	100	0.854
11.	*friedman_mse*	*Sqrt*	100	0.870
12.	*friedman_mse*	*log2*	100	0.870
13.	*absolute_error(mae)*	*Auto*	10	0.835
14.	*absolute_error(mae)*	*Sqrt*	10	0.857
15.	*absolute_error(mae)*	*log2*	10	0.857
16.	*absolute_error(mae)*	*Auto*	100	0.852
17.	*absolute_error(mae)*	*Sqrt*	100	0.871
18.	*absolute_error(mae)*	*log2*	100	0.871
19.	*Mse*	*Auto*	10	0.833
20.	*Mse*	*Sqrt*	10	0.852
21.	*Mse*	*log2*	10	0.852
22.	*Mse*	*Auto*	100	0.853
23.	*Mse*	*Sqrt*	100	0.870
24.	*mse*	*log2*	100	0.870

The Random forest regression R_Score value: 0.871 accuracy for both (mae,sqrt,100) & (mae,log2,100).

The finalised best saved model is **Support Vector Machine learning** model

R_Score value when compared to other model the accuracy is closer to 1

Accuracy

(0.877 rbf (Non linear and Hyper parameter C = 10000)