ASSIGNMENT-REGRESSION ALGORITHM

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

Based on the dataset of client required insurance charges

Machine learning-supervised learning-regression

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

Rows: 1338, Columns: 6

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

Sex (Male & female)- **dataset=pd.get_dummies(dataset,drop_first=True)** , which used to covert string to number

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.

Documented in Table:1,2,3

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

A) Random Forest Regressor (*min_samples_leaf=1*)

- **R^2 Score: 0.888** (highest among all models)
- **Prediction Accuracy:** Consistent and close to actual values

B) Why Chosen:

- Handles both linear and non-linear relationships.
- Reduces overfitting compared to Decision Trees.
- Outperformed SVR and Linear Regression in R^2 and practical prediction tests.

REGRESSION ASSIGNMENT

1.Multiple linear regression (r2 value= 0.789)

Table1. Support vector machine:

S.No	Hyper parameter	Poly (r value)	Sigmoid (r value)	Rbf (non linear r value)	Linear (r value)
1	C10	0.038	0.039	-0.032	0.462
2	C100	0.617	0.527	0.320	0.628
3	C500	0.826	0.444	0.664	0.763
4	C1000	0.856	0.287	0.810	0.764
5	C2000	0.860	-0.593	0.854	0.744
6	C3000	0.859		0.866	
7	C3500			0.869	

Table2:Decision Tree:

S.No	Criterion	Spliter	R value
1	mse	best	0.682
2	*squared_error*	best	**scikit-learn** does **not recognize**
3	friedman_mse	best	0.692
4	friedman_mse	*random*	0.736
5	*absolute_error*	random	**scikit-learn** does **not recognize**
6	poisson	best	**scikit-learn** does **not recognize**

Table3:Random Forest

S.No	Config	n_estimators	friedman_mse	*min_samples_leaf=1*	max_depth=None	min_samples_split=2
1	C10	0.833	0.833	**0.888**	0.845	0.865
2	C50	0.849	0.849	0.870	0.833	0.883
3	C100	0.853	0.853	0.728	0.833	0.872