



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:

Client want to predict the insurance charge based on the various inputs (age,bmi, how many children,gender,smoker or not)

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

1338 rows × 6 columns

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

Convert string to numerical for gender and smoker by using one hot encoding it is nominal)

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.

```
from sklearn.svm import SVR
regressor=SVR(kernel='poly',C=3000000)
regressor.fit(x_train,y_train)
```

Support vector machine r2 score = 0.8627905936051945

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

1.RANDOM FOREST

Si.no	criterion	Tuning parameters	R value
1.	squared_error	n_estimators=50	0.85
2.	squared_error	n_estimators=100	0.85
3.	absolute_error	n_estimators=100	0.85
4.	absolute_error	n_estimators=50	0.85
5.	friedman_mse	n_estimators=100	0.85
6.	friedman_mse	n_estimators=50	0.85
7.	poisson	n_estimators=100	0.85
8.	poisson	n_estimators=50	0.84
9.	poisson	max_depth=None	0.85
10.	squared_error	max_depth=None	0.85
11.	absolute_error	max_depth=None	0.85
12.	friedman_mse	max_depth=None	0.85
13.	absolute_error	Random_state=None	0.85
14.	squared_error	Random_state=None	0.85
15.	friedman_mse	Random_state=None	0.85
16.	poisson	Random_state=None	0.85
17.	friedman_mse	bootstrap=True	0.85
18.	poisson	bootstrap=True	0.85
19.	squared_error	bootstrap=True	0.85
20.	absolute_error	bootstrap=True	0.85

2.multi linear regression:

R score = 0.7894790349867009

3.support vector machine regression:

Si.no	kernel	Tuning parameter	R value
1.	linear	C=100	0.54
2.	linear	degree=3	-0.11
3.	linear	coef0=0.0	-0.11
4.	linear	gamma=scale	-0.11
5.	linear	gamma=auto	-0.11
6.	linear	C=3000	0.75
7.	linear	C=2000	0.68
8.	linear	C=1000	0.63
9.	rbf	C=1000	-0.11
10.	rbf	C=3000	-0.97
11.	sigmoid	C=100	-0.87
12.	poly	C=3000000	0.86

4.DECISSION TREE:

Si.no	criterion	splitter	R value
1.	friedman_mse	best	0.70
2.	absolute_error	best	0.65
3.	poisson	best	0.71
4.	squared_error	best	0.69
5.	squared_error	random	0.70
6.	friedman_mse	random	0.70
7.	poisson	random	0.71
8.	absolute_error	random	0.78

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

Best r2 value for each model:

1. multi linear regression R score = **0.7894790349867009**
2. **Support vector machine r2 score = 0.8627905936051945**
3. DECISSION TREE = **0.78**
4. RANDOM FOREST= **0.85**

After evaluating multiple models including Linear Regression, Decision Tree, Random Forest, and Support Vector Regression (SVR). I chose Support Vector Regression with a **polynomial kernel (C=3000000)** as the final model.

This model showed the best accuracy compared to the other algorithms.