**Assignments-Regression-Baby Step-3**

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. problem statement

Stage 1: Machine Learning

In this preliminary stage, the primary focus is on processing numerical data. Both input and output data are presented in numerical format, establishing the groundwork for subsequent stages.

Stage 2: Supervised Learning

With clearly defined requirements, this stage places a strong emphasis on providing explicit instructions and setting clear expectations.

Stage 1: Regression

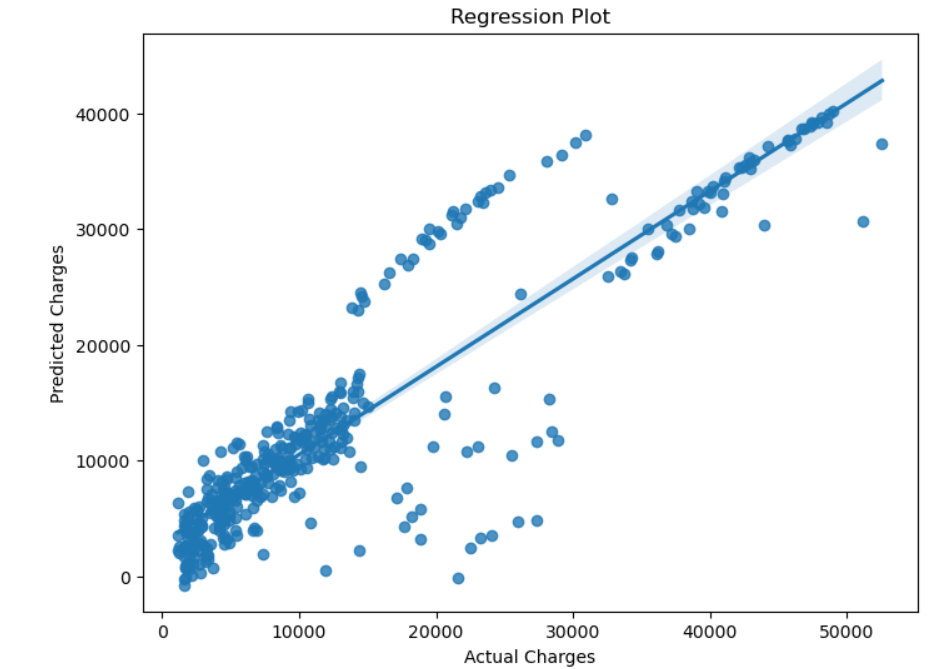
During this phase, the goal is to predict values for insurance charges based on various input features. The objective involves refining the model to produce accurate predictions, laying the foundation for subsequent stages in the machine learning process.

A sizable dataset, containing 1338 rows and 6 columns (age, sex, bmi, children, smoker, charges), underwent manipulation to facilitate analysis. Specifically, the categorical column "smoker" was transformed into numerical values using DataFrame modifications. Additionally, the "sex" column was converted into dummy variables, with the first column dropped to mitigate redundancy in the dummy data. Following these transformations, the dataset now comprises 1338 rows and 6 columns. Age, sex, bmi, children, and smoker function as independent variables (input), while charges serves as the dependent variable (output).

Subsequently, the data were divided into training and testing sets and subjected to analysis using multiple machine learning algorithms, namely multilinear regression, support vector machine (SVM), decision tree, and random forest. Various hyperparameter tuning parameters were employed in the process. The resulting values and outcomes are documented below.

Top of Form

**MULTILINEAR REGRESSION**



r\_score 0.7945694285037965

**DecisionTree**

**The decision tree Rgression R2 value for hyperparameter tuning parameters(criterion =(** poisson**), splitter=(** best**), max\_features=(** log2**) , R-squared = (** 0.777971**)**

|  |
| --- |
| **DecisionTree** |
| **sino criterion splitter max\_features R-squared** |
| 0 squared\_error best sqrt 0.669371 |
| 1 squared\_error best log2 0.704252 |
| 2 squared\_error random sqrt 0.685812 |
| 3 squared\_error random log2 0.677778 |
| 4 friedman\_mse best sqrt 0.596788 |
| 5 friedman\_mse best log2 0.748285 |
| 6 friedman\_mse random sqrt 0.672064 |
| 7 friedman\_mse random log2 0.675962 |
| 8 absolute\_error best sqrt 0.826151 |
| 9 absolute\_error best log2 0.715086 |
| 10 absolute\_error random sqrt 0.713686 |
| 11 absolute\_error random log2 0.693696 |
| 12 poisson best sqrt 0.674881 |
| 13 poisson best log2 0.777971 |
| 14 poisson random sqrt 0.695698 |
| 15 poisson random log2 0.592118 |

**RandomForest**

|  |
| --- |
| **RandomForest** |
| **Sino criterion n\_estimators max\_features** **R-squared** |
| 0 gini 10 sqrt 0.87043 |
| 1 gini 10 log2 0.87043 |
| 2 gini 100 sqrt 0.87043 |
| 3 gini 100 log2 0.87043 |
| 4 entropy 10 sqrt 0.87043 |
| 5 entropy 10 log2 0.87043 |
| 6 entropy 100 sqrt 0.87043 |
| 7 entropy 100 log2 0.87043 |
| 8 log\_loss 10 sqrt 0.87043 |
| 9 log\_loss 10 log2 0.87043 |
| 10 log\_loss 100 sqrt 0.87043 |
| 11 log\_loss 100 log2 0.87043 |
| 12 Mse 10 sqrt 0.87043 |
| 13 Mse 10 log2 0.87043 |
| 14 Mse 100 sqrt 0.87043 |
| 15 Mse 100 log2 0.87043 |

**The Random forest Rgression R2 value for hyperparameter tuning parameters(criterion =** gini**,** entropy, log\_loss , Mse **splitter =( 10,100), max\_features=** (log2, sqrt) **, R-squared =** 0.87043**)**

**SVM**

The SVM regression use R2-Value(nonlinear(rbf) and **hyperparameter tuning C** **(**10000)**=**0.886714

|  |
| --- |
| **SVM** |
| c linear poly rbf sigmoid |
| 10 0.500731 0.045912 -0.042389 0.042743 |
| 100 0.642332 0.659122 0.354710 0.535316 |
| 500 0.772336 0.845790 0.709443 0.442208 |
| 1000 0.750128 0.863186 0.828370 0.172030 |
| 2000 0.748534 0.864223 0.862534 -0.864681 |
| 10000 0.746695 0.861905 0.886714 -42.538914 |

CONCLUTION:

Among above modles **Random forest as highest R2-value**

**The Random forest Rgression R2 value for hyperparameter tuning parameters(criterion =** gini**,** entropy, log\_loss , Mse **splitter =( 10,100), max\_features=** (log2, sqrt) **, R-squared =** 0.87043**)**