***Requirement and Evaluation Metrics Analysis***

**Requirement:**

Client wants to predict insurance charges based on the several parameters. Client has provided the dataset for the same.

**1. Identify the problem statement.**

Client provided Inputs such as age, sex, BMI, no. of children, smoker or not.

We need to develop a best model to predict the insurance charges for these inputs in future.

**2. Basic info about the dataset.**

We have 5 inputs and 1 output column.

Age, BMI, and Children are integer columns, but Sex and Smoker columns are categorical.

Output is insurance charges, so it’s going to be number Which in turn means we need to perform Regression.

**3. Data Preprocessing:**

We need to verify the differences between the integer columns, if the difference is too high, we need to perform standardization.

We know Sex and Smoker Columns are categorical columns, so we need to verify whether it is ordinal or nominal data. (As we can understand by the nature of data that this is a Nominal data). So, we must perform One Hot Coding technique to convert our Nominal Data to numbers format.

Note: So proposed data preprocessing performed should be performed to the final concluded model in deployment phase as well.

Because all the models will be trained with preprocessed data and model understands only the preprocessed input format. So, we should remember to pass pre processed data in deployed model.

Here model’s Output will also be in standardized format. So, to get exact client’s output which is insurance charges we should reverse pre-process the model’s output.

**4. Develop a best model with high r2\_score.**

**Development Plan:**

**Stage 1**-->On understanding all inputs are numbered columns, we must choose Machine Learning Algorithms.

**Stage 2**-->Customer clearly knows what needs to be predicted, that is to predict the insurance charges, so under ML we must choose Supervised Learning.

**Stage 3**-->Since we understand output is numerical, we must choose Regression algorithms.

**5. All the research values of various model were documented below.**

Below Regression Algorithms were implemented and test results captured.

* Multiple Linear Regression
* Support Vector Machine SVM
* Decision Tree
* Random Forest

**Multiple Linear Regression:**

Here we got the r^2 score of 0.7865108093853883

**Support Vector Machine SVM:**

Below table holds the r^2 value captured for each Kernal available in SVM Regression and with different Hyper tuning parameters (C Value).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.No | Hyper  Parameter | RBF (NON-LINEAR) (r value) | POLY (r value) | SIGMOID (r value) |
| 1 | C10 | -0.048085569 | 0.027082271 | 0.01938609 |
| 2 | C500 | 0.639761904 | 0.815689631 | 0.463854986 |
| 3 | C1000 | 0.791561828 | 0.851982649 | 0.184221884 |
| 4 | C2000 | 0.846020871 | 0.857358931 | -0.578682233 |
| 5 | C3000 | **0.860998499** | 0.857789839 | -2.011925672 |

**Decision Tree:**

Below table holds the r^2 value for each criterion available in decision tree algorithm and with different Max\_Features and Splitter combinations.

This combination with highest r^2 value is highlighted.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.NO | CRETERION | MAX FEATURES | SPLITTER | R VALUE |
| 1 | squared\_error | AUTO | BEST | 0.69737097 |
| 2 | squared\_error | AUTO | RANDOM | 0.73443695 |
| 3 | squared\_error | SQRT | BEST | 0.67116164 |
| 4 | squared\_error | SQRT | RANDOM | 0.65982761 |
| 5 | squared\_error | LOG2 | BEST | 0.58434145 |
| 6 | squared\_error | LOG2 | RANDOM | 0.62537728 |
| 7 | friedman\_mse | AUTO | BEST | 0.69774707 |
| 8 | friedman\_mse | AUTO | RANDOM | 0.75065127 |
| 9 | friedman\_mse | SQRT | BEST | 0.756826 |
| 10 | friedman\_mse | SQRT | RANDOM | 0.67154568 |
| 11 | friedman\_mse | LOG2 | BEST | 0.65467339 |
| 12 | friedman\_mse | LOG2 | RANDOM | 0.58923556 |
| 13 | absolute\_error | AUTO | BEST | 0.69863458 |
| 14 | absolute\_error | AUTO | RANDOM | 0.7557315 |
| 15 | absolute\_error | SQRT | BEST | 0.67593411 |
| 16 | absolute\_error | SQRT | RANDOM | 0.68021871 |
| 17 | absolute\_error | LOG2 | BEST | 0.68158755 |
| 18 | absolute\_error | LOG2 | RANDOM | 0.72919265 |
| 19 | poisson | AUTO | BEST | 0.65420683 |
| 20 | poisson | AUTO | RANDOM | 0.69994438 |
| 21 | poisson | SQRT | BEST | 0.6741052 |
| 22 | poisson | SQRT | RANDOM | 0.70242058 |
| 23 | poisson | LOG2 | BEST | 0.62445977 |
| 24 | poisson | LOG2 | RANDOM | 0.69959111 |

**Random Forest:**

Below table holds the r^2 values captured using Random Forest Algorithm with different n\_estimators and highest r^2 score is achieved here and highlighted.

We have a tie at absolute\_error with SQRT and LOG2 with n\_estimators=100, so I would lean to choose the SQRT Feature because SQRT transformation is often more interpretable in a wide range of applications and less sensitive to very small values or zeros in the data.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **CRETERION** | **MAX FEATURES** | **n\_estimators** | **RF\_R^2 Value** |
| 1 | squared\_error | AUTO | 50 | 0.857755749 |
| 2 | squared\_error | AUTO | 100 | 0.860130384 |
| 3 | squared\_error | SQRT | 50 | 0.868441302 |
| 4 | squared\_error | SQRT | 100 | 0.872237376 |
| 5 | squared\_error | LOG2 | 50 | 0.868441302 |
| 6 | squared\_error | LOG2 | 100 | 0.872237376 |
| 7 | friedman\_mse | AUTO | 50 | 0.856879107 |
| 8 | friedman\_mse | AUTO | 100 | 0.859468257 |
| 9 | friedman\_mse | SQRT | 50 | 0.86751624 |
| 10 | friedman\_mse | SQRT | 100 | 0.871683106 |
| 11 | friedman\_mse | LOG2 | 50 | 0.86751624 |
| 12 | friedman\_mse | LOG2 | 100 | 0.871683106 |
| 13 | absolute\_error | AUTO | 50 | 0.856240956 |
| 14 | absolute\_error | AUTO | 100 | 0.859200853 |
| 15 | absolute\_error | SQRT | 50 | 0.874568503 |
| 16 | absolute\_error | SQRT | 100 | 0.876483781 |
| 17 | absolute\_error | LOG2 | 50 | 0.874568503 |
| 18 | absolute\_error | LOG2 | 100 | 0.876483781 |
| 19 | poisson | AUTO | 50 | 0.830332835 |
| 20 | poisson | AUTO | 100 | 0.83277911 |
| 21 | poisson | SQRT | 50 | 0.832442804 |
| 22 | poisson | SQRT | 100 | 0.837648154 |
| 23 | poisson | LOG2 | 50 | 0.832442804 |
| 24 | poisson | LOG2 | 100 | 0.837648154 |

**Conclusion:**

As we can observe from the above model’s estimation metrics, we can conclude that random forest with n\_estimators=75 has the high-performance r^2 score.

Based on this we can conclude that this is the best model to predict the insurance charges.