**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.

Data set: <https://github.com/Kuppusamy104/Machine-Learning/blob/main/Assignment-Regression%20Algorithm/insurance_pre.csv>

**1. Identify your problem statement:**

Predict insurance charges based on input parameters (age, sex, BMI, children, smoker).

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

* Total Rows :1338
* Total Coolum’s : 6
* Input/ Independent : age, sex, BMI, children, smoker
* Output / Dependent : charges (The End goal is predict the charges )

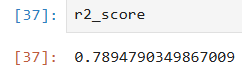
**3.) Mention the pre-processing method if you’re doing any (like converting string to number – nominal data)**

* Data set contains categorical data for two columns (sex, smoker)
* **Sex** : Male / Female –comes under **Nominal data**
* **Smoker :** Yes / or - comes under **Nominal data**
* **Using One-hot encoding** method to converts categorical data (Sex,Smoker) into numerical values to understand machine learning algorithms to create a model.
* **Input/Independent** : 5 columns - age, bmi, children, charges, sex\_male, smoker\_yes
* **Output/ dependent :** One column – charges

**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.**

I have created the below machine learning regression algorithm to predict best r2 values with various parameter. The Random Forest Model is provided good accuracy compare with other model.

* Multiple Linear Regression R2 value is: **0.789479**



* SVM- Support Vector Machine R2 values:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Hyper Parameter** | **Kernel =Linear**  **R2 score** | **Kernel= RBF(Non linear)  R2Score** | **Kernel= POLY**  **R2 Score** | **Kernel=SIGMOID**  **R2 Score** |
| 1 | C 1.0 | -0.0101026 | -0.083382 | -0.075699 | -0.07542 |
| 2 | C 10 | 0.4624684 | -0.03227 | 0.03871622 | 0.0393071 |
| 3 | C 100 | 0.628879 | 0.3200317 | 0.617956 | 0.527610 |
| 4 | **C 1000** | 0.7649311 | 0.810206 | **0.8566487** | 0.287470 |

* Decision Tree

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Criterion** | **Max Features** | **Splitter** | **R2 Score** |
| 1 | Squared\_error | None | best | 0.6958 |
| **2** | **squared\_error** | **sqrt** | **best** | **0.7622** |
| 3 | squared\_error | log2 | best | 0.6605 |
| 4 | squared\_error | None | random | 0.7173 |
| 5 | squared\_error | sqrt | random | 0.6514 |
| 6 | squared\_error | log2 | random | 0.7370 |
| 7 | friedman\_mse | None | best | 0.6973 |
| 8 | friedman\_mse | sqrt | best | 0.7006 |
| 9 | friedman\_mse | log2 | best | 0.7182 |
| 10 | friedman\_mse | None | random | 0.6989 |
| 11 | friedman\_mse | sqrt | random | 0.6644 |
| 12 | friedman\_mse | log2 | random | 0.6578 |
| 13 | absolute\_error | None | best | 0.6422 |
| 14 | absolute\_error | sqrt | best | 0.6435 |
| 15 | absolute\_error | log2 | best | 0.6777 |
| 16 | absolute\_error | None | random | 0.6804 |
| 17 | absolute\_error | sqrt | random | 0.5990 |
| 18 | absolute\_error | log2 | random | 0.6815 |
| 19 | poisson | None | best | 0.7280 |
| 20 | poisson | sqrt | best | 0.746 |
| 21 | poisson | log2 | best | 0.7212 |
| 22 | poisson | None | random | 0.7375 |
| 23 | poisson | sqrt | random | 0.5816 |
| 24 | poisson | log2 | random | 0.6653 |

* Random Forest

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Criterion** | **Max Features** | **n\_estimators** | **R2 Score** |
| 1 | Squared\_error | None | 50 | 0.8498 |
| 2 | squared\_error | sqrt | 50 | 0.8695 |
| 3 | squared\_error | log2 | 50 | 0.8695 |
| 4 | squared\_error | None | 100 | 0.8538 |
| 5 | squared\_error | sqrt | 100 | 0.87102 |
| 6 | squared\_error | log2 | 100 | 0.87102 |
| 7 | friedman\_mse | None | 50 | 0.85007 |
| 8 | friedman\_mse | sqrt | 50 | 0.87024 |
| 9 | friedman\_mse | log2 | 50 | 0.87024 |
| 10 | friedman\_mse | None | 100 | 0.85405 |
| 11 | friedman\_mse | sqrt | 100 | 0.871054 |
| 12 | friedman\_mse | log2 | 100 | 0.871054 |
| 13 | absolute\_error | None | 50 | 0.85266 |
| 14 | absolute\_error | sqrt | 50 | 0.870814 |
| 15 | absolute\_error | log2 | 50 | 0.87081 |
| 16 | absolute\_error | None | 100 | 0.85200 |
| **17** | **absolute\_error** | **sqrt** | **100** | **0.87106858** |
| **18** | **absolute\_error** | **log2** | **100** | **0.87106858** |
| 19 | poisson | None | 50 | 0.84910 |
| 20 | poisson | sqrt | 50 | 0.8632 |
| 21 | poisson | log2 | 50 | 0.86323 |
| 22 | poisson | None | 100 | 0.8526 |
| 23 | poisson | sqrt | 100 | 0.86801 |
| 24 | poisson | log2 | 100 | 0.8680 |

**5.) All the research values (r2\_score of the models) should be documented. (You can make tabulation or screenshot of the results.)**

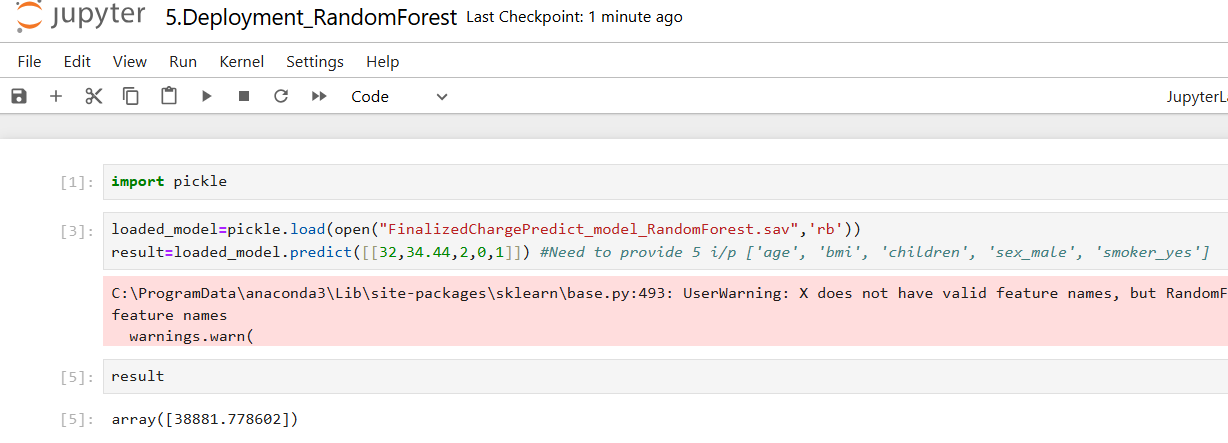
Please refer point 4.

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

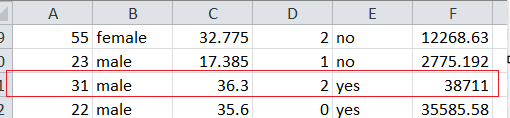
**Finally the best model is:** **Random Forest**, Random Forest regression algorithm using the parameter settings: criterion=" absolute\_error " , max\_features =”sqrt”, n\_estimators=100 & random\_state=0.

This indicates that approximately **87.10%** of the variance in the target variable is explained by the model with these settings.

**7.) Deployment:**

****

**8.) Manual comparison with dataset providing nearest input, The output is matching closely.**

****