**Final Documentation - Insurance charges prediction**

1. **Problem statement identification**

According to the requirements, the goal is to predict insurance charges based on several parameters.

**Stage 1: Domain Selection**

* Machine Learning is chosen since the dataset primarily contains numerical data.

**Stage 2: Learning Selection**

* **Supervised learning** is appropriate because:

- The requirement is clearly defined (predicting insurance charges).

- Both input features and output labels are available.

**Stage 3: Supervised Learning Type**

* Since the output label (insurance charges) consists of numerical values, the problem falls under **Regression**.

ML -> Supervised -> Regression

1. **Basic information about the given dataset**

The objective is to predict insurance charges using the features Age, Sex, BMI, Children, and Smoker. The dataset consists of 1338 rows and 6 columns.

**3. Pre-processing methods**The Sex and Smoker columns are nominal data without order, so they are converted into numbers using One-Hot Encoding.

**4. Find the good model with r2\_score**

**Machine Learning Algorithms:**

**Simple Linear Regression->** Not suitable, as the dataset contains multiple input features rather than a single input.

**Multiple Linear Regression ->** Applied to the dataset, resulting in an R² score of 0.7891.

**Support Vector Machine Regression (Non-linear):**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hyper parameter** | **linear**  **(r value)** | **poly**  **(r value)** | **rbf**  **(r value)** | **sigmoid**  **(r value)** |
| C=10 | -0.0017 | -0.0930 | -0.0818 | -0.0909 |
| C=100 | 0.5432 | -0.0992 | -0.1245 | -0.1185 |
| C=500 | 0.6269 | -0.0817 | -0.1245 | -0.4735 |
| C=1000 | 0.6338 | -0.0546 | -0.1176 | -1.7112 |
| C=2000 | 0.6898 | -0.0016 | -0.1078 | -5.8190 |
| C=3000 | 0.7590 | 0.0494 | -0.0962 | -12.5445 |

The SVM Regression use R2 Value(linear and hyper parameter (C=3000)) = 0.7590

**Decision Tree Regression:**

|  |  |  |  |
| --- | --- | --- | --- |
| **criterion** | **max\_features** | **splitter** | **R Value** |
| friedman\_mse | log2 | random | 0.64632 |
| friedman\_mse | log2 | best | 0.7163 |
| friedman\_mse | sqrt | random | 0.6230 |
| friedman\_mse | sqrt | best | 0.7496 |
| squared\_error | log2 | random | 0.6605 |
| squared\_error | log2 | best | 0.7769 |
| squared\_error | sqrt | random | 0.6397 |
| squared\_error | sqrt | best | 0.7119 |
| absolute\_error | log2 | random | 0.7367 |
| absolute\_error | log2 | best | 0.6030 |
| absolute\_error | sqrt | random | 0.7589 |
| absolute\_error | sqrt | best | 0.6187 |
| poisson | log2 | random | 0.6611 |
| poisson | log2 | best | 0.6911 |
| poisson | sqrt | random | 0.6510 |
| poisson | sqrt | best | 0.6527 |

The Decision Tree Regression use R2 value (criterian=squared\_error, max\_features=log2 and splitter=best) = 0.7769

**Random Forest Regression:**

|  |  |  |  |
| --- | --- | --- | --- |
| **criterion** | **max\_features** | **n\_estimators** | **R Value** |
| friedman\_mse | log2 | 10 | 0.8568 |
| friedman\_mse | log2 | 100 | 0.8632 |
| friedman\_mse | sqrt | 10 | 0.8547 |
| friedman\_mse | sqrt | 100 | 0.8665 |
| squared\_error | log2 | 10 | 0.8419 |
| squared\_error | log2 | 100 | 0.8644 |
| squared\_error | sqrt | 10 | 0.8594 |
| squared\_error | sqrt | 100 | 0.8682 |
| poisson | log2 | 10 | 0.8435 |
| poisson | log2 | 100 | 0.8612 |
| poisson | sqrt | 10 | 0.8500 |
| poisson | sqrt | 100 | 0.8671 |

The Random Forest Regression use R2 value (criterian=squared\_error, max\_features=sqrt and n\_estimators=100) = 0.8682

1. **The final model for machine learning best method of Regression:**

**Random Forest R2 Value (squared\_error, sqrt, 100) = 0.8682**

The Random Forest algorithm was chosen as it provides results that closely approach 1 for a perfect model.