**Assignment’s problem statement**

A client 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.

* **Identify the problem statement**

After studied this problem statement clearly, I found three stages of model screening process, after this process we can tell, how the model should be for this assignment

**Stage 1: (“Domain selection”)**

* Our dataset in numbers and our predicted column also in numbers so the machine learning domain gives best model among others.

**Stage 2: (“Learning selection”)**

* We have clear input and output so we will go with supervised learning

**Stage 3: (“Find classification or regression”)**

* Our output is numerical so that its regression
* **Tell basic info about the dataset**

Well, our dataset has **1338 rows and 6 columns**, mostly in numbers except the column name sex (Male, Female)

* **Pre-processing method**

Pre-processing is needed in our dataset because sex column has **nominal data**. So, we should convert this nominal data to number by using **one hot encoding method**

* **Final good model**

if r2\_score near 1 we consider that best model, after compare other models with respective algorithms Support vector machine gives best model with r2\_score 0.8734

* **Comparison between different algorithms** 
  + 1. Algorithm – Simple \_Linear – regression – **Our dataset has multiple inputs so we can’t use this algorithm**
    2. Algorithm – Multilinear – regression - R2\_score – 0.7894
    3. Algorithm – Support vector machine – regression - R2\_score 0.7590
    4. Algorithm – Decision tree – regression - R2\_score -0.7660
    5. Algorithm – Random \_Forest – regression - R2\_score – 0.8734

Following tabulation shows that r2\_score with combination of respective fine turn hyper parameters

Algorithm – Support vector machine – Support vector regression

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Si.no | Penalty value (c) | Linear r2\_score | RBF  (nonlinear) | Poly r2\_score | Sigmoid r2\_score |
| 1 | C=1.0 | -0.1116 | -0.088 | -0.0642 | -0.0899 |
| 2 | C=100 | 0.5432 | -0.1248 | -0.0099 | -0.1181 |
| 3 | C=1000 | 0.6340 | -0.1174 | -0.0555 | - |
| 4 | C=2000 | 0.6893 | -0.1077 | -0.0027 | - |
| 5 | C=3000 | 0.7590 | -0.0962 | 0.0489 | - |

Note: penalty value too high leads to overfitting

Best r2\_score is – 0.7590

Algorithm – Decision tree – Decision tree regression

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Si.no | Criterion | Splitter | Max\_  features | Min\_impurity  (Float=0-1) | Ccp-alpha  (Float=0-infiniti) | R2\_score |
| 1. | Mse | Best | None | 0.0 | 0.0 | 0.6895 |
| 2. | Mse | Random | 3 | 0.0 | 0.01 | 0.6926 |
| 3. | Mse | Best | Sqrt | 0.1 | 0.0 | 0.7325 |
| 4. | Mse | Best | 3 | 0.01 | 0.05 | 0.7277 |
| 5. | Mse | Random | None | 0.1 | 0.05 | 0.7660 |
| 6. | Mse | Random | None | 0.05 | 0.1 | 0.6730 |
| 7. | Mse | Best | Log2 | 0.05 | 0.1 | 0.7540 |
| 8. | Friedman\_mse | Random | Sqrt | 0.01 | 0.01 | 0.6680 |
| 9. | Friedman\_mse | Best | None | 0.0 | 0.0 | 0.6877 |
| 10. | Friedman\_mse | Random | Log2 | 0.05 | 0.05 | 0.5899 |
| 11. | Friedman\_mse | Random | None | 0.0 | 0.01 | 0.7294 |
| 12. | Friedman\_mse | Best | 4 | 0.1 | 0.0 | 0.6280 |
| 13. | Mae | Best | Log2 | 0.05 | 0.1 | 0.6593 |
| 14. | Mae | Random | 5 | 0.01 | 0.1 | 0.7284 |
| 15. | Mae | Random | Sqrt | 0.01 | 0.0 | 0.7473 |
| 16. | Mae | Best | None | 0.0 | 0.1 | 0.6716 |
| 17. | Mae | Best | 2 | 0.1 | 0.05 | 0.7386 |
| 18. | Mae | Best | 3 | 0.1 | 0.0 | 0.7005 |

Note: Mse- mean squered error

Mae- mean absolute error

Best r2\_score is – 0.7660

Algorithm – Random \_Forest – regression

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Si.no | N\_Estimators | Criterion | Max\_  features | Random\_state | R2\_score |
| 1. | 50 | Mse | None | 0.0 | 0.8498 |
| 2. | 100 | Mse | None | 0.0 | 0.8538 |
| 3. | 100 | Mse | Sqrt | 0.0 | 0.8710 |
| 4. | 50 | Mse | Sqrt | 42 | 0.8727 |
| 5. | 100 | Mse | Log2 | 0.0 | 0.8710 |
| 6. | 50 | Mse | None | 42 | 0.8569 |
| 7. | 100 | Mse | Log2 | 42 | 0.8734 |
| 8. | 50 | Friedman\_mse | None | 0.0 | 0.8500 |
| 9. | 50 | Friedman\_mse | Sqrt | 0.0 | 0.8702 |
| 10. | 50 | Friedman\_mse | None | 42 | 0.8560 |
| 11. | 50 | Friedman\_mse | Log2 | 42 | 0.8721 |
| 12 | 50 | Mae | None | 0.0 | 0.8526 |
| 13 | 50 | Mae | Log2 | 0.0 | 0.8708 |
| 14 | 50 | Mae | Sqrt | 0.0 | 0.8708 |
| 15 | 50 | Mae | None | 42 | 0.85035 |
| 16 | 50 | Poisson | None | 0.0 | 0.84910 |
| 17 | 50 | Poisson | Sqrt | 0.0 | 0.86323 |
| 18 | 50 | Poisson | Log2 | 42 | 0.87104 |

Note: Mse- mean squared error

Mae- mean absolute error

Best r2\_score is – 0.8734

**Conclusion**

**After working with all Machine learning algorithms now we can compare and identify good model with fine tune hyper parameters. (**Note: if r2\_score is near 1 its consider as a good model )

**The good model and r2\_value of this dataset is -** Algorithm – Random \_Forest – Best r2\_score is- 0.8734