**To find the best suitable model for the given Problem Statement and to predict insurance charges**

**Dataset**

- The provided dataset comprises 6 columns and 1338 rows.
- Two rows in a dataset are of the string data type and are transformed to integers.

## Multiple Regression

| **Model** | **r_score** |
|---|---|
| **Multiple linear regression** | 0.7890995064322818 |

## Support vector machine

| **s.no** | **Hyper parameter** | **Linear** | **Rbf** | **Poly** | **Sigmoid** |
|---|---|---|---|---|---|
| 1 | **C=0.1** | -0.12207668380229886 | -0.08957624598812952 | -0.08625251710262294 | -0.08974351910465961 |
| 2 | **C = 1 (default value)** | -0.11166128719608448 | -0.08842732776913875 | -0.06429258402105531 | -0.0899412170256757 |
| 3 | **C = 10** | -0.0016176324886472138 | -0.08196910396420853 | -0.09311615532848516 | -0.09078319814614 |
| 4 | **C = 100** | 0.5432818196692804 | -0.12480367775039669 | -0.09976172333666167 | -0.11814554828411405 |
| 5 | **C= 500** | 0.6270462757743913 | -0.1246416131929442 | -0.082028798630986 | -0.45629443405234804 |
| 6 | **C = 1000** | 0.634036931263208 | -0.11749092439183229 | -0.055505937517909665 | -1.6659081315533064 |
| 7 | **C = 2000** | 0.6893263105100382 | -0.10778764037675015 | -0.0027024512793158983 | -5.6164315417244275 |

**The support vector machine's highest r_score value 0.68932105100382 is using a linear hyperparameter C=0.1.**

## Decision Tree

| s.no | Criterion | Splitter | max_features | r_score |
|---|---|---|---|---|
| 1 | Squared_Error | best | auto | 0.715198014 |
| 2 | | | sqrt | 0.726308394 |
| 3 | | | log2 | 0.665649362 |
| 4 | | random | auto | 0.649502166 |
| 5 | | | sqrt | 0.678818261 |
| 6 | | | log2 | 0.594698239 |
| 7 | Friedman_Mse | best | auto | 0.705572742 |
| 8 | | | sqrt | 0.725020357 |
| 9 | | | log2 | 0.718273632 |
| 10 | | random | auto | 0.680074312 |
| 11 | | | sqrt | 0.667404571 |
| 12 | | | log2 | 0.655036223 |
| 13 | Absolute_Error | best | auto | 0.726618812 |
| 14 | | | sqrt | 0.632390368 |
| 15 | | | log2 | 0.691638872 |
| 16 | | random | auto | 0.705281874 |
| 17 | | | sqrt | 0.748569956 |
| 18 | | | log2 | 0.684427514 |
| 19 | Poisson | best | auto | 0.723928106 |
| 20 | | | sqrt | 0.685875931 |
| 21 | | | log2 | 0.423392503 |
| 22 | | random | auto | 0.725945925 |
| 23 | | | sqrt | 0.442143058 |
| 24 | | | log2 | 0.684386122 |

- **The Decision Tree's highest r_score value is** 0.748569956 **using hyperparameter** Criterion = Squared_Error , Splitter= **random, max_features=auto**

## Random Forest

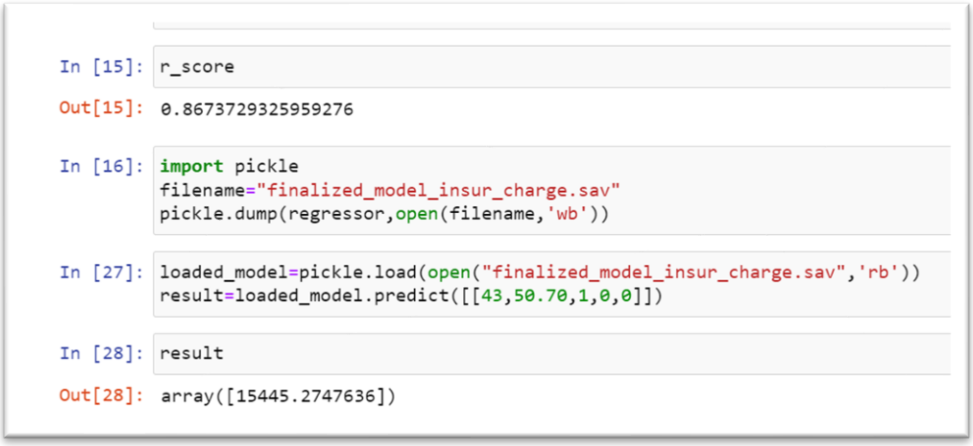
| s.no | criterion | n_estimators | max_features | r_score |
|---|---|---|---|---|
| 1 | squared_error | 10 | sqrt | 0.85113292 |
| 2 | | | log2 | 0.85113292 |
| 3 | | | auto | 0.813275595 |
| 4 | | 50 | sqrt | 0.867046344 |
| 5 | | | log2 | 0.867046344 |
| 6 | | | auto | 0.833810287 |
| 7 | | 100 | sqrt | 0.867372933 |
| 8 | | | log2 | 0.867372933 |
| 9 | | | auto | 0.838443585 |
| 10 | absolute_error | 10 | sqrt | 0.845650549 |
| 11 | | | log2 | 0.845650549 |
| 12 | | | auto | 0.822887257 |
| 13 | | 50 | sqrt | 0.859194771 |
| 14 | | | log2 | 0.859194771 |
| 15 | | | auto | 0.83653449 |
| 16 | | 100 | sqrt | 0.861781537 |
| 17 | | | log2 | 0.861781537 |
| 18 | | | auto | 0.840116123 |
| 19 | friedman_mse | 10 | sqrt | 0.851453708 |
| 20 | | | log2 | 0.851453708 |
| 21 | | | auto | 0.813696974 |
| 22 | | 50 | sqrt | 0.867260894 |
| 23 | | | log2 | 0.867260894 |
| 24 | | | auto | 0.833417218 |
| 25 | | 100 | sqrt | 0.867012385 |
| 26 | | | log2 | 0.867012385 |
| 27 | | | auto | 0.838707448 |
| 28 | poisson | 10 | sqrt | 0.846653477 |
| 29 | | | log2 | 0.846653477 |
| 30 | | | auto | 0.811882035 |
| 31 | | 50 | sqrt | 0.860580004 |
| 32 | | | log2 | 0.860580004 |
| 33 | | | auto | 0.835304119 |
| 34 | | 100 | sqrt | 0.862760007 |
| 35 | | | log2 | 0.862760007 |
| 36 | | | auto | 0.839250721 |

- **The Random forest's highest r_score value is 0.867372933 using hyperparameter Criterion = squared_error, n_estimators=100, max_features=sqrt**

## Conclusion

| S.No | Model | r_score |
|---|---|---|
| 1 | multiple linear regression | 0.7890995064322818 |
| 2 | support vector machine | 0.6893263105100382 |
| 3 | decision tree | 0.748569956 |
| **4** | **random forest** | 0.867372933 |

As a result, **random forest** is the finalised and has the **greatest r_score value**.



```
In [15]: r_score
Out[15]: 0.8673729325959276

In [16]: import pickle
         filename="finalized_model_insur_charge.sav"
         pickle.dump(regressor,open(filename,'wb'))

In [27]: loaded_model=pickle.load(open("finalized_model_insur_charge.sav",'rb'))
         result=loaded_model.predict([[43,50.70,1,0,0]])

In [28]: result
Out[28]: array([15445.2747636])
```