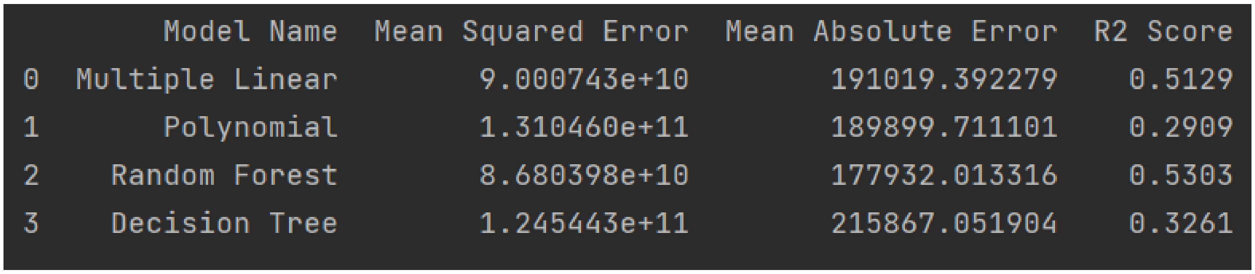
Software Components

- Module_evaluator package.
    - This is the parent folder for the project. It contains two other packages: regression and classification
- Regression package
    - This folder has three different modules namely, base_regression.py, evaluate.py and models.py
        - Base_regression.py – This module contains a function called fix_missing_data and a regression Base class
            - Base Class – contains all methods that are common to all defined regression models
            - Fix-missing-data Function – fixes missing data by finding the mean of the columns with missing data and replacing the missing data with the calculated means.
        - Evaluate.py – contains two functions: perform_regression and evaluate_regression.
            - Perform_regression - receives regression model and test_size as parameters and then goes through the steps of performing the supplied regression.
            - Evaluate_regression – checks to make sure that the data file path provided to be evaluated exists ,instanciates the various regression models and then calls the perform regression function and returns the resulting data frame as output.
        - Models.py – contains all the regression models (as classes) upon which our dataset will be evaluated. It inherits the base class from the base_regression.py module.
- Classification package
    - Like the regression package, this folder has three different modules: base_classification.py, evaluate.py and models.py.
        - Base_classification.py – This module contains a function called fix_missing_data and a classification Base class
            - Base Class – contains all methods that are common to all defined classifictoin models
            - Fix-missing-data Function – does the same thing as the function in the base_regresson.py module.
        - Evaluate.py – contains two functions: perform_classification and evaluate_classification.
            - Perform_classification - receives classification model and test_size as parameters and then goes through the steps of performing the supplied classification algorithm.

    - Evaluate_classification – checks to make sure that the data file path provided to be evaluated exists ,instanciates the various classification models and then calls the perform_classification function and returns the resulting data frame as output.
- Models.py – contains all the classification models (as classes) upon which our dataset will be evaluated. It inherits the base class from the base_classification.py module.

The Output

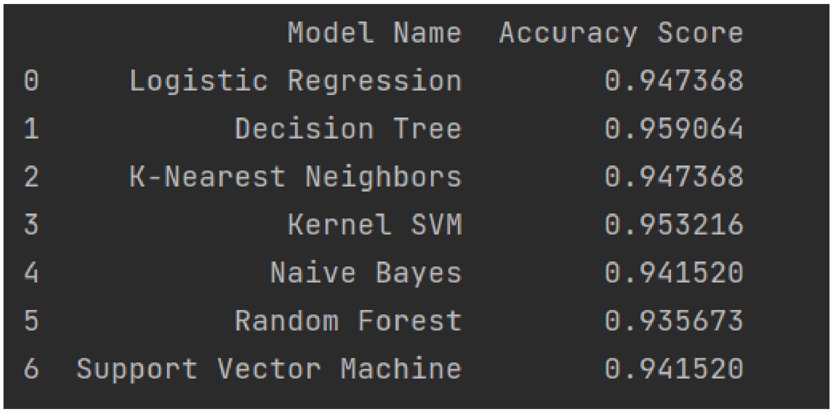
Regression selection



```
        Model Name  Mean Squared Error  Mean Absolute Error  R2 Score
0  Multiple Linear        9.000743e+10        191019.392279    0.5129
1       Polynomial        1.310460e+11        189899.711101    0.2909
2    Random Forest        8.680398e+10        177932.013316    0.5303
3    Decision Tree        1.245443e+11        215867.051904    0.3261
```

For the supplied dataset, the model with the maximum R2 score is best suited to be used. In this case Random Forest.

Classification selection



```
               Model Name  Accuracy Score
0     Logistic Regression        0.947368
1           Decision Tree        0.959064
2     K-Nearest Neighbors        0.947368
3              Kernel SVM        0.953216
4             Naive Bayes        0.941520
5           Random Forest        0.935673
6  Support Vector Machine        0.941520
```

The decision tree model has the maximum accuracy score hence it should be the ideal model for the supplied dataset.