PROBLEM STATEMENT :

The problem is to develop a predictive model that can accurately predict insurance charges based on several parameters provided in the dataset.

SPECIFICATIONS OF DATASET:

**Total number of rows: 1338**

**Total number of columns: 6**

There are six columns in the dataset:

1. age: Represents the age of the individual (Number - integer).
2. sex: Represents the gender of the individual (String).
3. bmi: Represents the Body Mass Index of the individual (Number - double).
4. children: Represents the number of children the individual has (Number - integer).
5. smoker: Indicates whether the individual is a smoker or not (String).
6. charges: Represents the insurance charges for the individual (Number - double).

PREPROCESSING:

We need to handle categorical variables like "sex" and "smoker," which are currently represented as strings. Machine learning algorithms typically require numerical input, so we need to convert these categorical variables into numerical format. Here are the pre-processing methods for handling these categorical variables:

1. Converting "sex" to numerical format: Since "sex" has two categories (e.g., male and female), we can use binary encoding to convert it into numerical representation. For example:

* Male: 1
* Female: 0

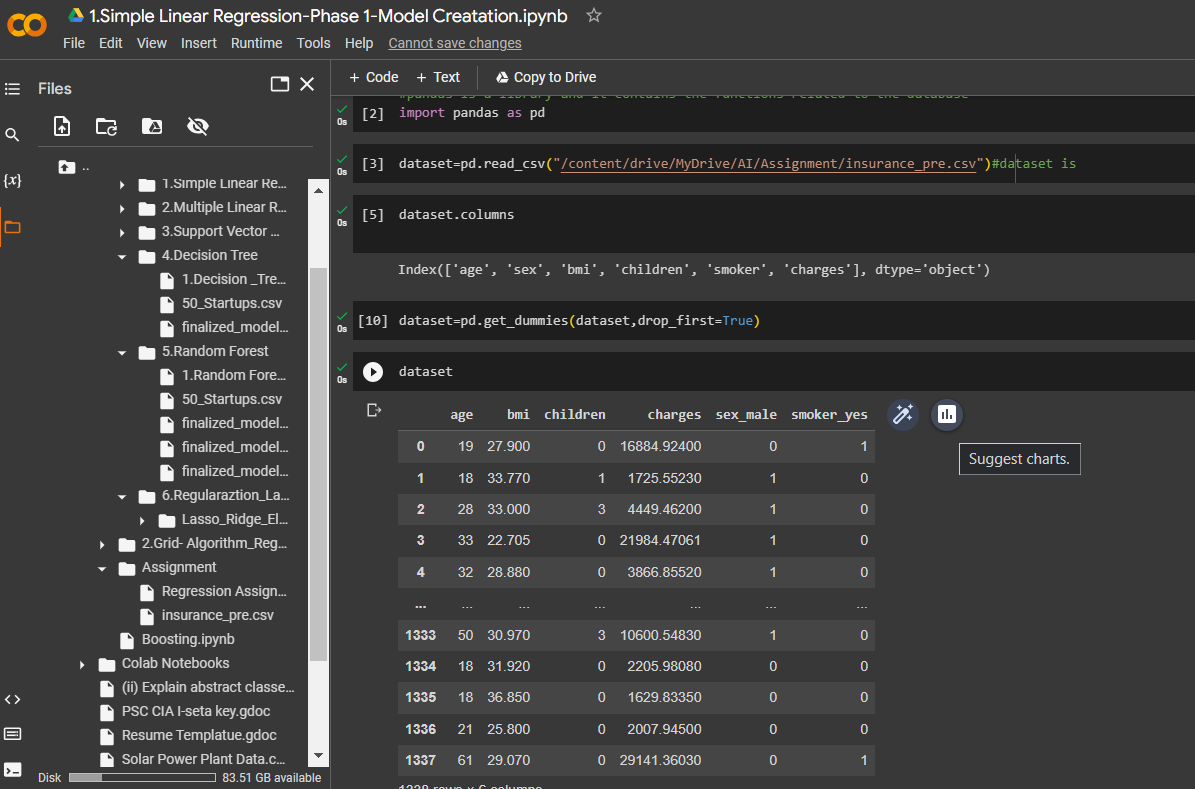
If there were more than two categories, we might use one-hot encoding to create binary columns for each category.

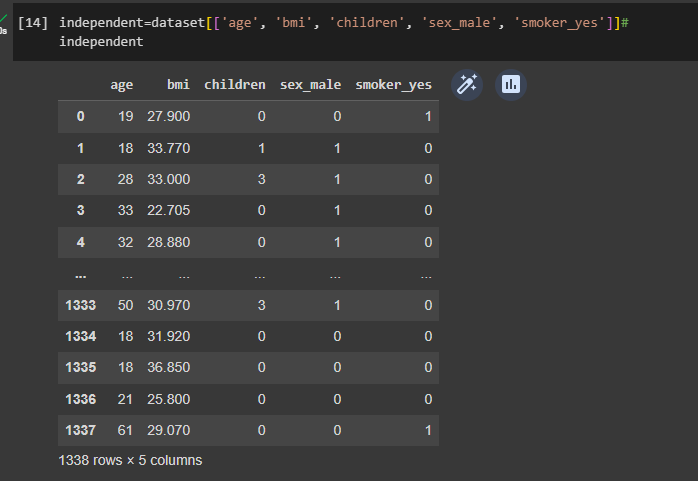
1. Converting "smoker" to numerical format: Similarly, for the "smoker" variable, which has two categories (e.g., smoker and non-smoker), we can use binary encoding:

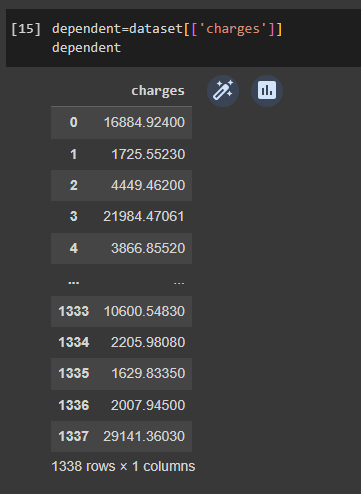
* Smoker: 1
* Non-smoker: 0

Again, if there were more than two categories, we might use one-hot encoding.

After converting these categorical variables into numerical format, the dataset will have all the features represented as numerical values, and it will be ready for building the predictive model.



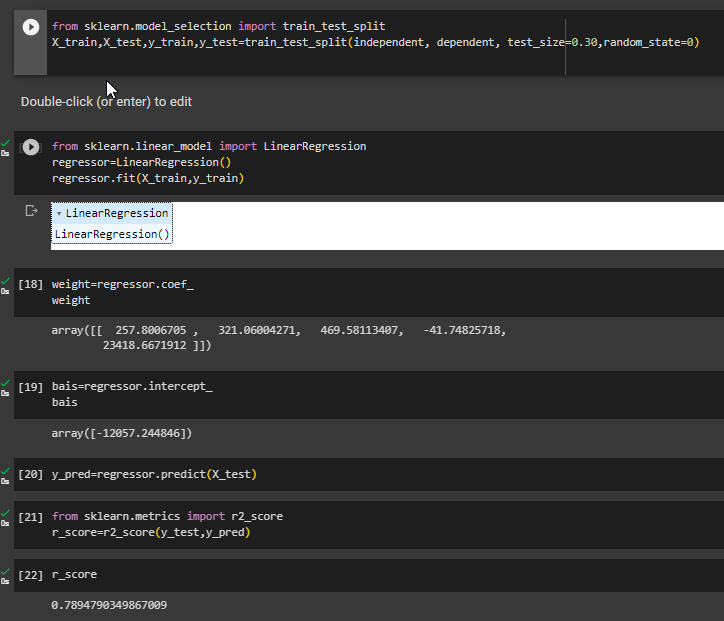
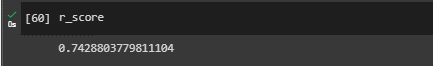




MODEL EVALUATION RESULTS:

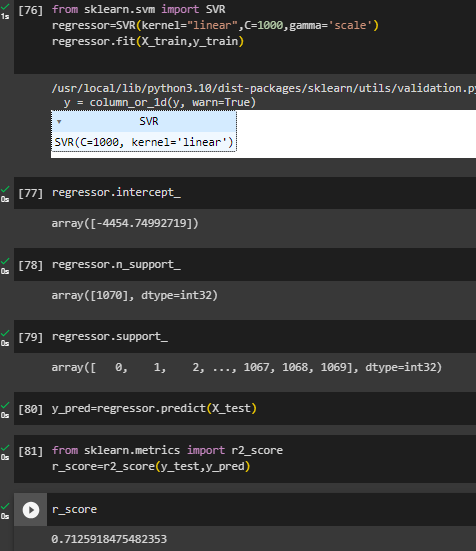
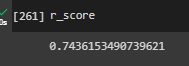
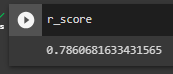
1) SIMPLE LINEAR REGRESSION

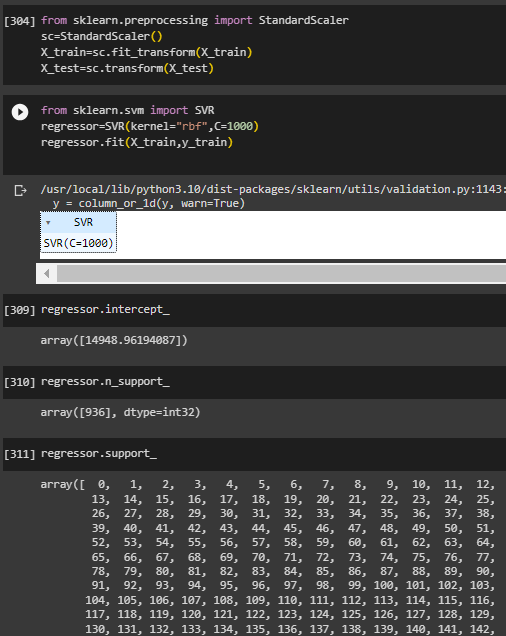
|  |  |
| --- | --- |
| **Parameters** | R^2 |
| test\_size=0.30,random\_state=0 | 0.78 |
| test\_size=0.20,random\_state=2 | 0.74 |

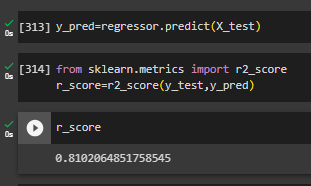
  
   


2) Support Vector Regression.

|  |  |
| --- | --- |
| Parameters | R^2 |
| kernel="linear",C=1000,gamma='scale | 0.71 |
| kernel="linear",C=100,gamma='scale' | 0.54 |
| (kernel="linear",C=100000,gamma='scale' | 0.74 |
| kernel="linear",C=1000000,gamma='scale' | 0.78 |
|  |  |



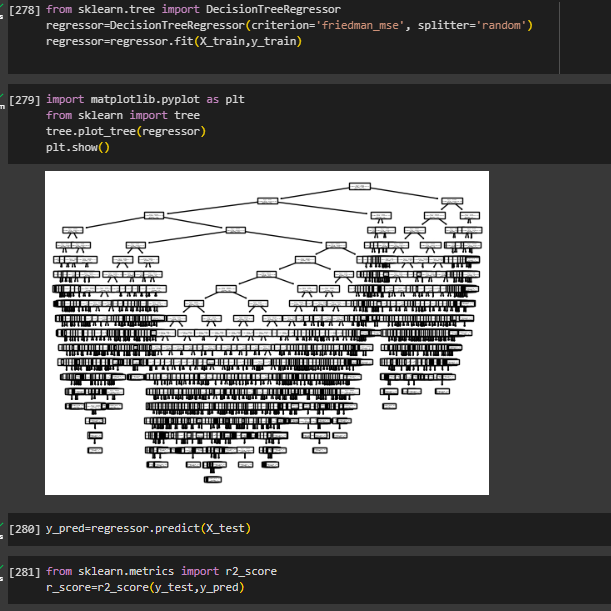
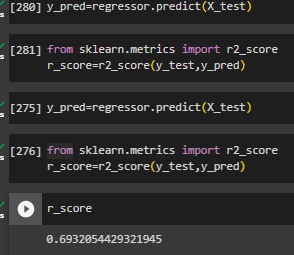
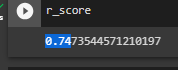
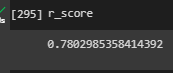
WITH STANDARD SCALAR   


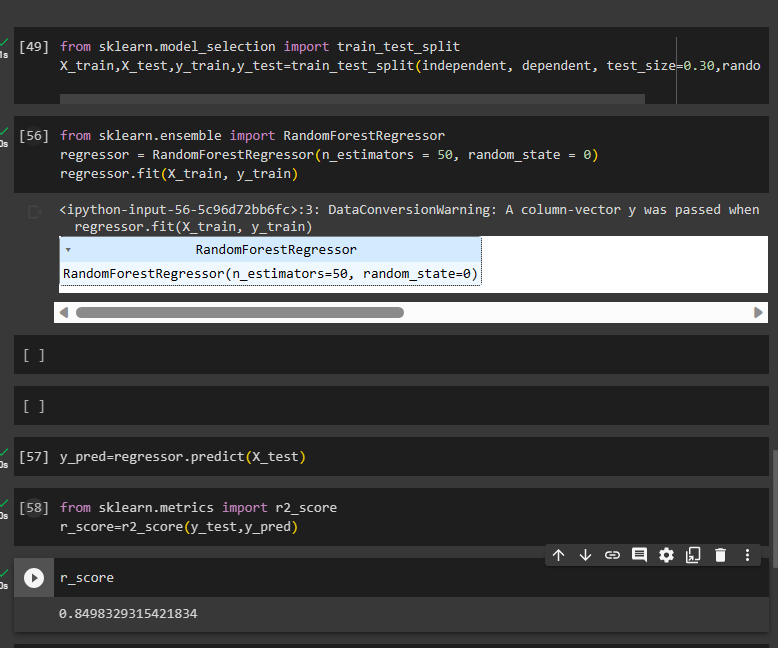


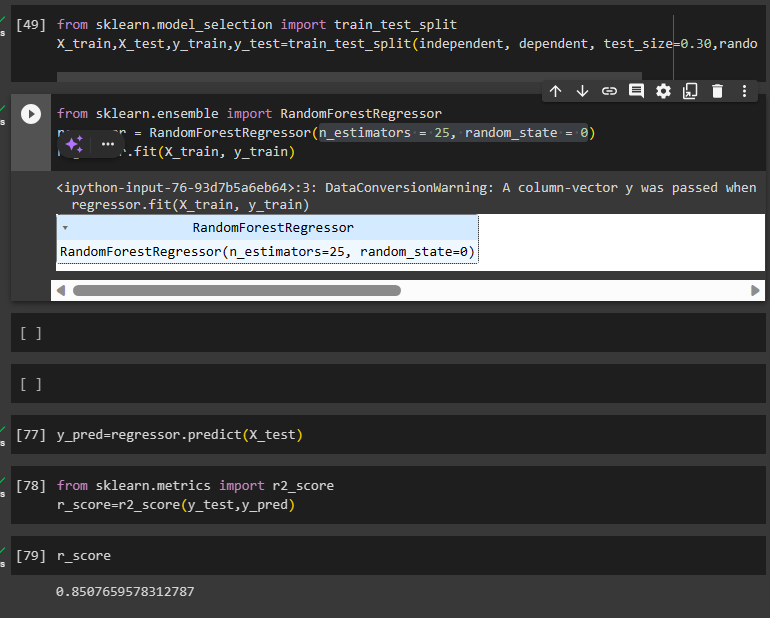
|  |  |
| --- | --- |
| Parameters | R^2 |
| kernel="rbf",C=1000 | 0.81 |
| kernel="linear",C=100,gamma='scale' | 0.54 |
| (kernel="linear",C=100000,gamma='scale' | 0.74 |

3) DECISION TREE

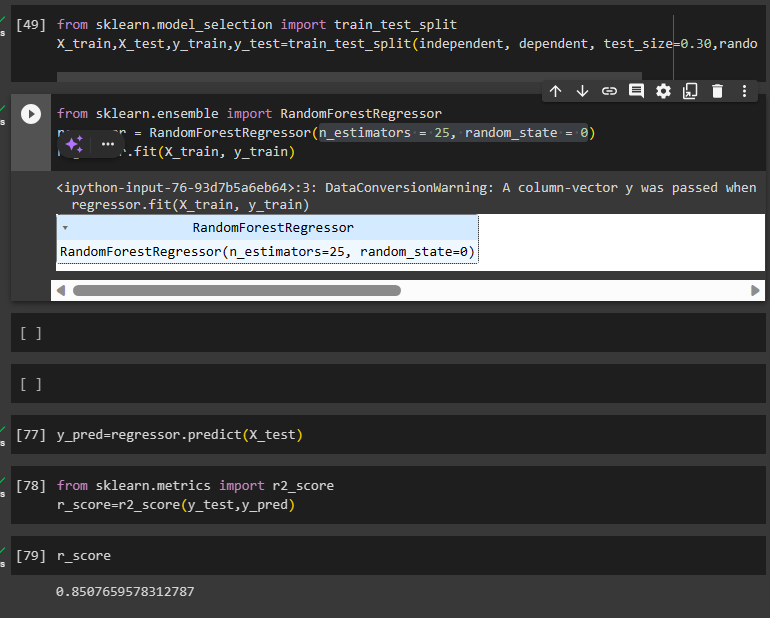
|  |  |
| --- | --- |
| Parameters | R^2 |
| criterion='friedman\_mse', splitter='random | 0.69 |
| criterion='poisson', splitter='random' | 0.74 |
| criterion='absolute\_error', splitter='random' | 0.78 |

4) RANDOM FOREST  




BEST MODEL:  
 RANDOM FOREST REGRESSOR



regressor = RandomForestRegressor(n\_estimators = 25, random\_state = 0)

BEST ACCURACY : R2 : 0.85