Assignment-Regression Algorithmwww.hopelearning.netHope [AIadmin@hopelearning.net](mailto:AIadmin@hopelearning.net)

Download Set:Google Drive Link Click here.Git Hub Link: <https://raw.githubusercontent.com/RamishaRaniK/dataset/main/insurance_pre.csv>

Problem Statement or Requirement:

A client’s requirement is, he wants to predict the insurance charges based on the several parameters. The Client has provided the dataset of thesame.

As a data scientist, you mustdevelop amodel which will predict the insurance charges.

1.)Identify your problem statement

***Ans: Model need to predict the insurance charges for the user’s input.***

2.)Tell basic info about the dataset (Total number of rows, columns)

***Dataset has 1339 rows and 6 columns***

3.)Mention the pre-processing method if you’re doing any (like converting string to number –nominal data)

***To convert the categorical column to Nominal column, one hot encoding is used.***

4.)Develop a good model with r2\_score.You can use any machine learning algorithm; you can create many models.Finally, you have to come up with final model.

***Random forest is the best model with r2\_score=0.8714662234663751***

5.)All the research values (r2\_score of the models) should be documented. (You can make tabulation or screenshot of the results.)

**Answer:**

**Multiple\_Linear\_Regression** – **R2\_value = 0.7865108093853883**

**SUPPORT VECTOR MACHINE:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.no** | **Hyper Parameter** | **Linear**  **(r\_Value)** | **RBF(Non-Linear)**  **(r\_value)** | **POLY**  **(r\_value)** | **SIGMOID**  **(r\_value)** |
| 1 | C=10 | -0.04030337559469621 | -0.09675799768426963 |  | -0.0987122675639922 |
| 2 | C=100 | 0.5218991729199273 | -0.08849432786729228 |  | -0.0987122675639922 |
| 3 | C=500 | 0.6133408106776344 | -0.07682733093111382 |  | -0.0987122675639922 |
| 4 | C=1000 | 0.6188266977660104 | -0.06746998827970918 |  | -0.0987122675639922 |
| 5 | C=2000 | 0.6253680212857904 | -0.028555725655149633 |  | -0.0987122675639922 |
| 6 | C=3000 | 0.6662613638532224 | 0.012370799372016905 |  | -0.0987122675639922 |

**Decision Tree**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.no** | **Criterion** | **Max Features** | **Splitters** | **R-Value** |
| 1 | mse | auto’ | 'best' | 0.6803989539447486 |
| 2 | mse | 'auto' | 'random' | 0.6665369275476931 |
| 3 | mse | sqrt | 'best' | 0.7154300792370711 |
| 4 | mse | sqrt | 'random' | 0.6573942950427881 |
| 5 | mse | Log2 | 'best' | 0.7160147240172816 |
| 6 | mse | Log2 | 'random' | 0.6254976524703708 |
| 7 | mae | Log2 | 'best' | 0.7341611535777841 |
| 8 | mae | Log2 | 'random' | 0.6993710881310722 |
| 9 | mae | sqrt | 'best' | 0.6659468434136084 |
| 10 | mae | sqrt | 'random' | 0.7370011480340668 |
| 11 | 'mae' | 'auto' | 'best' | 0.6805637551461692 |
| 12 | 'mae' | 'auto' | 'random' | 0.7430248530403654 |
| 13 | ‘friedman\_mse’ | ‘auto’ | ‘best’ | 0.679573770920941 |
| 14 | ‘friedman\_mse’ | ‘auto’ | 'random' | 0.6162857789641702 |
| 15 | ‘friedman\_mse’ | sqrt | best’ | 0.7073943870407575 |
| 16 | ‘friedman\_mse’ | sqrt | 'random' | 0.7070529505926102 |
| 17 | ‘friedman\_mse’ | Log2 | best’ | 0.6122382005109666 |
| 18 | ‘friedman\_mse’ | Log2 | 'random' | 0.7498422259058333 |

**Random Forest**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.no** | **Criterion** | **Max Features** | **n\_estimators** | **R-Value** |
| 1 | mse | auto’ | 10 | 0.8413507422450529 |
| 2 | mse | 'auto' | 100 | 0.8547489374661243 |
| 3 | mse | sqrt | 10 | 0.8466824295279097 |
| 4 | mse | sqrt | 100 | 0.8721562256156744 |
| 5 | mse | Log2 | 10 | 0.8716791016496523 |
| 6 | mse | Log2 | 100 | 0.8683850854756227 |
| 7 | mae | 'auto' | 10 | 0.8546970702599246 |
| 8 | mae | 'auto' | 100 | 0.8580953581969537 |
| 9 | mae | sqrt | 10 | 0.8649489755485535 |
| 10 | mae | sqrt | 100 | 0.8742121278998023 |
| 11 | 'mae' | Log2 | 10 | 0.8530203301863101 |
| 12 | 'mae' | Log2 | 100 | 0.8714662234663751 |

6.)Mention your final model, justify why u have chosen the same.

**Random Forest is the final model**

**from sklearn.ensemble import RandomForestRegressor**

**regressor=RandomForestRegressor(criterion='mae', max\_features=log2, n\_estimators=100)**

**regressor.fit(X\_train,y\_train)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Criterion** | **Max Features** | **n\_estimators** | **R-Value** |
| 'mae' | Log2 | 100 | 0.8714662234663751 |