**To Predict the Chronic Disease Using Linear Classificaiton Algorithm**

**To Find the Best Algorithm based on the R2 Score Value**

1. **) SUPPORT VECTOR MACHINE**

Using support vector machine algorithm, we achieved the R2 score value as **R2 Score : 0.9**

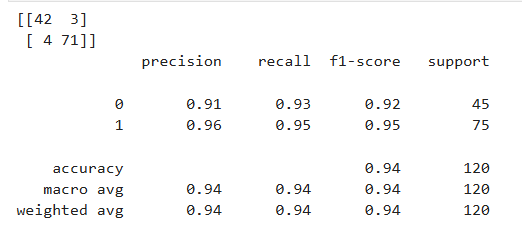
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SNO** | **HYPER**  **PARAMETER** | **LINEAR**  **(R VALUE)** | **RBF**  **(R VALUE)** | **POLY**  **(R VALUE)** | **SIGMOID**  **(R VALUE)** |
| 1 | C10 | -0.04 | -0.06 | -0.05 | -0.05 |
| 2 | C100 | 0.11 | -0.05 | -0.02 | -0.03 |
| 3 | C500 | 0.59 | -0.02 | 0.11 | 0.07 |
| 4 | C1000 | 0.78 | 0.01 | 0.27 | 0.19 |
| 5 | C2000 | 0.88 | 0.07 | 0.48 | 0.4 |
| 6 | C3000 | 0.9 | 0.12 | 0.64 | 0.59 |

1. **)DECISION TREE:**

Using Decision Tree algorithm, we got the best model with these hyper tuning parameters

BEST MODEL : {'criterion': 'log\_loss', 'max\_features': 'log2', 'splitter': 'random'}

|  |  |  |  |
| --- | --- | --- | --- |
| **SL\_NO** | **CRITERION** | **MAX FEATURES** | **SPLITTER** |
| 1 | GINI | SQRT | BEST |
| 2 | GINI | SQRT | RANDOM |
| 3 | GINI | LOG2 | BEST |
| 4 | GINI | LOG2 | RANDOM |
| 5 | ENTROPY | SQRT | BEST |
| 6 | ENTROPY | SQRT | RANDOM |
| 7 | ENTROPY | LOG2 | BEST |
| 8 | ENTROPY | LOG2 | RANDOM |
| 9 | LOG\_LOSS | SQRT | BEST |
| 10 | LOG\_LOSS | SQRT | RANDOM |
| 11 | LOG\_LOSS | LOG2 | BEST |
| 12 | LOG\_LOSS | LOG2 | RANDOM |

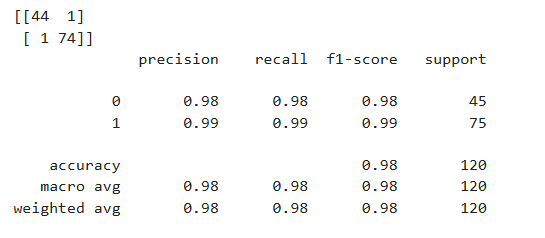


1. **RANDOM FOREST:**

Using Random Forest algorithm, we got the best model with the hyper tuning parameters

BEST MODEL : {'criterion': 'log\_loss', 'max\_features': 'log2', 'splitter': 'random'}

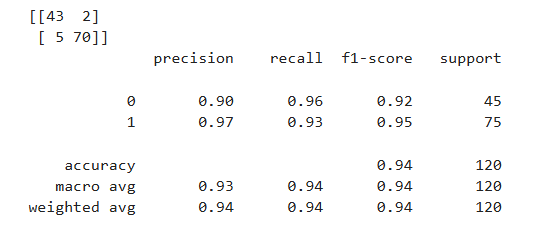
|  |  |  |  |
| --- | --- | --- | --- |
| **SL\_NO** | **CRITERION** | **MAX FEATURES** | **N\_ESTIMATORS** |
| 1 | ENTROPY | NONE | 50 |
| 2 | ENTROPY | NONE | 100 |
| 3 | ENTROPY | NONE | 250 |
| 4 | ENTROPY | NONE | 500 |
| 5 | ENTROPY | NONE | 1000 |
| 6 | ENTROPY | LOG2 | 50 |
| 7 | ENTROPY | LOG2 | 100 |
| 8 | ENTROPY | LOG2 | 250 |
| 9 | ENTROPY | LOG2 | 500 |
| 10 | ENTROPY | LOG2 | 1000 |
| 11 | ENTROPY | SQRT | 50 |
| 12 | ENTROPY | SQRT | 100 |
| 13 | ENTROPY | SQRT | 250 |
| 14 | ENTROPY | SQRT | 500 |
| 15 | ENTROPY | SQRT | 1000 |
| 16 | GINI | NONE | 50 |
| 17 | GINI | NONE | 100 |
| 18 | GINI | NONE | 250 |
| 19 | GINI | NONE | 500 |
| 20 | GINI | NONE | 1000 |
| 21 | GINI | LOG2 | 50 |
| 22 | GINI | LOG2 | 100 |
| 23 | GINI | LOG2 | 250 |
| 24 | GINI | LOG2 | 500 |
| 25 | GINI | LOG2 | 1000 |
| 26 | LOG\_LOSS | NONE | 50 |
| 27 | LOG\_LOSS | NONE | 100 |
| 28 | LOG\_LOSS | NONE | 250 |
| 29 | LOG\_LOSS | NONE | 500 |
| 30 | LOG\_LOSS | NONE | 1000 |
| 31 | LOG\_LOSS | LOG2 | 50 |
| 32 | LOG\_LOSS | LOG2 | 100 |
| 33 | LOG\_LOSS | LOG2 | 250 |
| 34 | LOG\_LOSS | LOG2 | 500 |
| 35 | LOG\_LOSS | LOG2 | 1000 |



1. **LOGISTIC REGRESSION MODEL:**

Using Logistic Regression algorithm, we got the best model with the hyper tuning parameters

BEST MODEL : {C : 1.0}



1. **KNN MODEL:**

Using Logistic Regression algorithm, we got the best model with the hyper tuning parameters

BEST MODEL : {'algorithm': 'auto', 'metric': 'minkowski', 'n\_neighbors': 10}

|  |  |  |  |
| --- | --- | --- | --- |
| **SL\_NO** | **METRIC** | **ALGORITHM** | **N\_NEIGHBORS** |
| 1 | MINKOWSKI | AUTO | 5 |
| 2 | MINKOWSKI | AUTO | 10 |
| 3 | MINKOWSKI | AUTO | 100 |
| 4 | MINKOWSKI | AUTO | 150 |
| 5 | MINKOWSKI | AUTO | 200 |
| 6 | MINKOWSKI | BALL\_TREE | 5 |
| 7 | MINKOWSKI | BALL\_TREE | 10 |
| 8 | MINKOWSKI | BALL\_TREE | 100 |
| 9 | MINKOWSKI | BALL\_TREE | 150 |
| 10 | MINKOWSKI | BALL\_TREE | 200 |
| 11 | MINKOWSKI | KD\_TREE | 5 |
| 12 | MINKOWSKI | KD\_TREE | 10 |
| 13 | MINKOWSKI | KD\_TREE | 100 |
| 14 | MINKOWSKI | KD\_TREE | 150 |
| 15 | MINKOWSKI | KD\_TREE | 200 |
| 16 | MINKOWSKI | BRUTE | 5 |
| 17 | MINKOWSKI | BRUTE | 10 |
| 18 | MINKOWSKI | BRUTE | 100 |
| 19 | MINKOWSKI | BRUTE | 150 |
| 20 | MINKOWSKI | BRUTE | 200 |

