**Fraud Detection or Not Assignment**

**Problem Statement:**

To create a best model for fraud detection or not based on given input parameters

**Basic info about the dataset:**

16426 rows \* 9 columns

Index(['step', 'type', 'amount', 'oldbalanceOrg', 'newbalanceOrig',

'oldbalanceDest', 'newbalanceDest', 'isFraud', 'isFlaggedFraud'],

dtype='object')

**Pre-processing Method:**

Using One Hot Encoding, we are converting “type” column from string to number. Dropping first column

Index(['step', 'amount', 'oldbalanceOrg', 'newbalanceOrig', 'oldbalanceDest',

'newbalanceDest', 'isFraud', 'isFlaggedFraud', 'type\_CASH\_OUT',

'type\_DEBIT', 'type\_PAYMENT', 'type\_TRANSFER'],

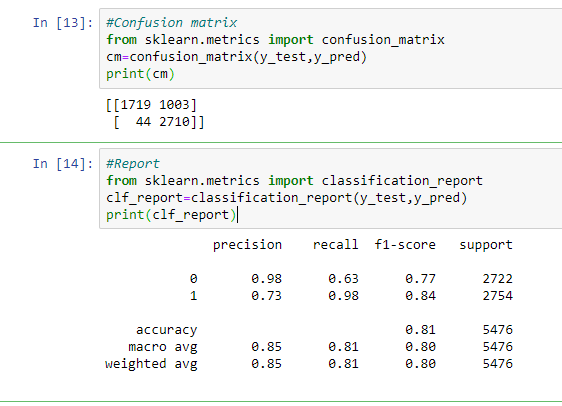
dtype='object')

**Model:**

1. Logistic Regression
2. Navies Bayes
3. KNN
4. Decision Tree
5. Random Forest
6. SVM

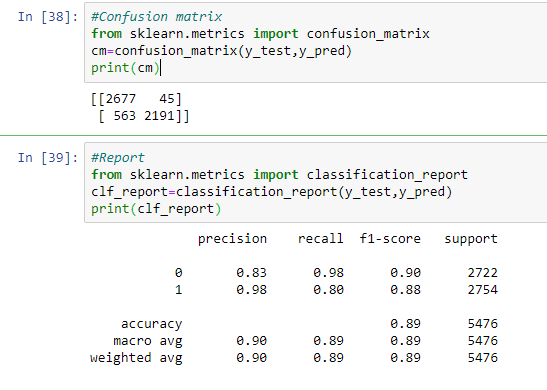
**Screenshot:**

**Logistic Regression:**

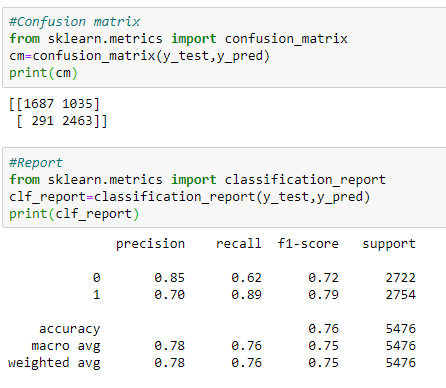


**Navies Bayes:**

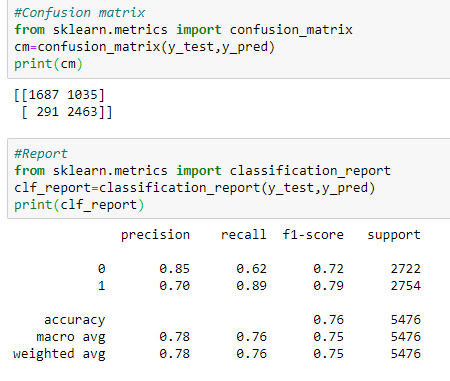
**GaussianNB**



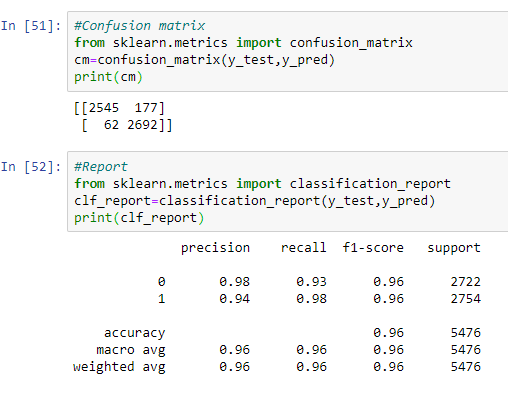
**MultinomialNB**



**ComplementNB**

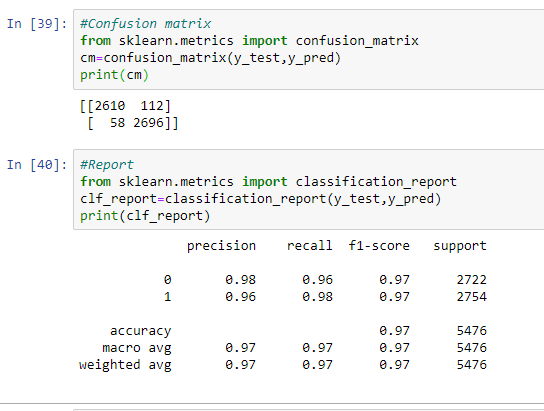


**BernoulliNB**

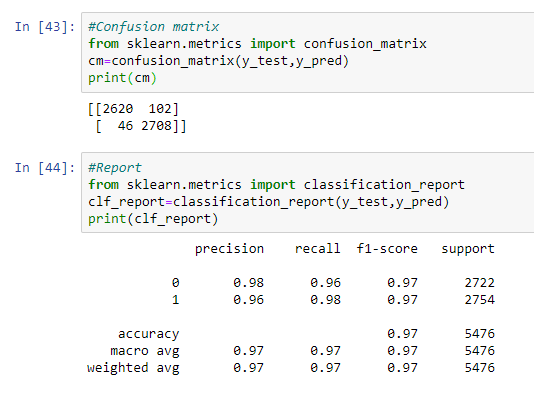


**KNN**

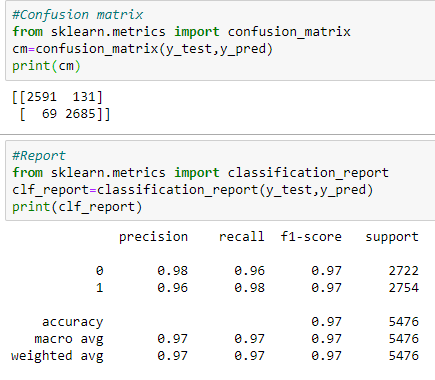
**N=7**



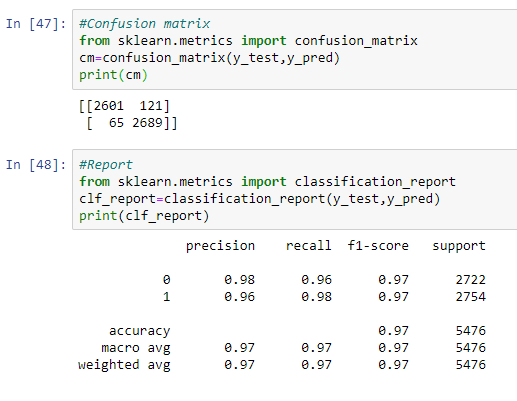
**N=3**



**N=11**

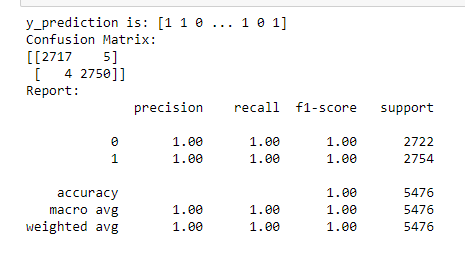


**N=9**

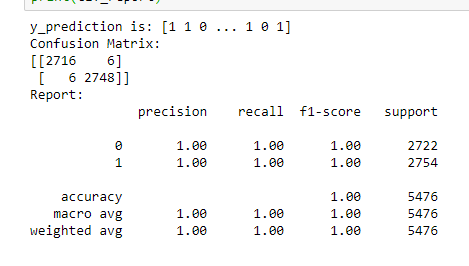


**Decision Tree:**

**Criterion= entropy**

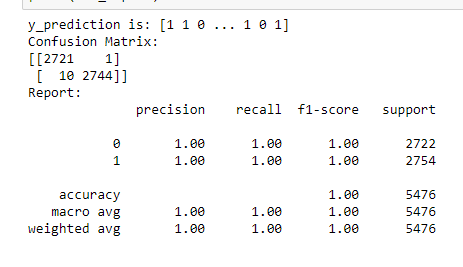


**Criterion= gini**



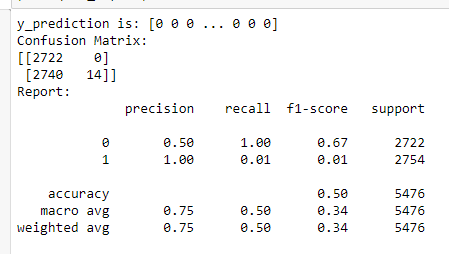
**Random Forest:**

**n\_estimators = 10, criterion = entropy**



**SVM:**

**Kernel=rbf**



**Final Model & Justification:**

Random Forest and Decision Tree has accuracy = 1.0

However, type 1 error for Random Forest is 1 and type 1 error for Decision Tree is 5

Hence, **Random Forest is the best model for the Fraud Detection or Not** Problem statement