CSE342: SML Course Project Report

Shobhit Raj (2022482)
Computer Science & Engineering Dept.
IIIT-Delhi, India
shobhit22482@iiitd.ac.in

Vashu (2022606)
Computer Science & Artificial Intelligence Dept.
IIIT-Delhi, India
vashu22606@iiitd.ac.in

I. Introduction

The rise of the internet and e-commerce appears to entail the usage of online payment transactions. Every second, thousands of deals are completed on the internet trading platform. The increased usage of online payments is leading to a rise in fraud. Financial fraud poses a significant threat to the integrity of FinTech banking transactions, leading to substantial financial losses and erosion of trust among customers. Fraud detection is an important component of online payment systems since it serves to protect both customers and merchants from financial damages. In this project, we propose a fraud detection system for online payments that uses machine learning techniques to identify and prevent fraudulent transactions. By developing effective fraud detection methods, we aim to safeguard financial transactions, protect customer assets, and maintain the trust and confidence of users in FinTech platforms. Our approach strives to improve fraud detection accuracy while reducing the amount of false positives, resulting in a more efficient and effective method for identifying and combating fraud.

Problem Statement: "Detecting and preventing financial fraud in FinTech banking transactions using Statistical Machine Learning techniques."

II. Literature Review

The Online Transaction Fraud Detection System has been the subject of several studies. We studied these before beginning the project in order to comprehend the numerous approaches that have been employed in the past. Bhattacharyya et al. note that while fraud algorithms are actively used by banks and payment companies, the breadth of studies on the use of machine learning techniques for payment fraud detection is limited [1], possibly due to the sensitive nature of the data. Their study concluded that random forests, though not widely deployed, may outperform more traditional methods. In his research, S. P. Maniraj concentrated on pre-processing and analyzing data sets and also using various algorithms like anomaly detection on the PCA-translated Credit Card Transaction data, including Isolation Forest and Local Outlier, and found that they perform better than the supervised learning models. [2] A similar research domain was presented by Wen-Fang YU and Na Wang where they applied outlier detection mining and distance sum algorithms to effectively estimate fraudulent transactions in an experimental data set from a particular commercial bank. [3] Outlier mining is mostly employed in the financial and online industries. They contain attributes that were taken from consumer behavior and based on the value of those attributes, they have estimated the gap between the observed value of that attribute and its actual value.

III. Dataset Details

We have used two datasets for training the models.
1. JP Morgan Dataset
2. PaySim Dataset

Firstly, the synthetic-data used in the project is especially provided by JP Morgan for research purposes which replicate the intricacies of real transactional data. For this project, we will be working with a comprehensive dataset that encompasses transactional data from fintech banking transactions. The dataset would include a wide range of features such as transaction amount, transaction type, time of transaction, sender and receiver's demographics, etc. Additionally, the dataset would contain labels indicating whether each transaction is fraudulent or legitimate. It contains 12 parameters and nearly 15,00,000 banking transactions labeled as either legitimate or fraudulent.

Second, we have used the Kaggle dataset "Synthetic Financial Datasets For Fraud Detection" generated by the PaySim mobile money simulator. The dataset would include a wide range of features such as transaction amount, transaction type, time of transaction, sender and receiver's bank account details, etc. This dataset has 10 parameters and nearly 6,00,000 transactions.

Objectives with the dataset –
1. Analyze different patterns in the dataset and discern complex patterns
2. Exploring the nuances of the dataset with severe class imbalance

A. Dataset Preprocessing and Cleaning Steps

We started by loading our dataset from the csv file into a DataFrame named 'finance_dataset'. For each column with missing values we imputed the values in the column assigning majorly using mode, mean and "unknown". For handling categorical data, we applied label encoding to several columns using 'LabelEncoder' from scikit-learn. Additionally, we removed the 'Transaction_Id' column as it was a unique identifier not useful for our model training. we also removed 'Time_step' column.

For the PaySim dataset, we loaded the dataset from the csv file into a DataFrame named ‘dataset‘. We observed that there are no null values in the dataset. Then, we started exploring the dataset (EDA), like the transaction types, no. of fraud cases, etc. We found that the fraudulent transactions are only of 2 types, ‘Transfer‘ & ‘Cash_Out‘. Remarkably, the number of fraudulent ‘Transfer‘ almost equals the number of fraudulent ‘Cash_out‘. Hence, we can conclude that fraud is committed by first transferring out funds to another account which subsequently cashes it out. Also, ‘isFraud‘ is always set when ‘isFlaggedFraud‘ is set, so can drop this feature as it is insignificant in predicting fraud transactions.

Lastly, we split the dataset into features (‘X‘) and labels (‘Y‘), with ‘Y‘ containing the ’Label’ column indicating fraud or not fraud, to prepare for our machine learning model training. We also split our dataset into training and testing sets using the ‘train_test_split‘ function, resulting in ‘X_Train‘, ‘Y_Train‘, ‘X_Test‘, and ‘Y_Test‘, to evaluate our models’ performance on unseen data.

B. Dataset Visualizations - JP Morgan Dataset

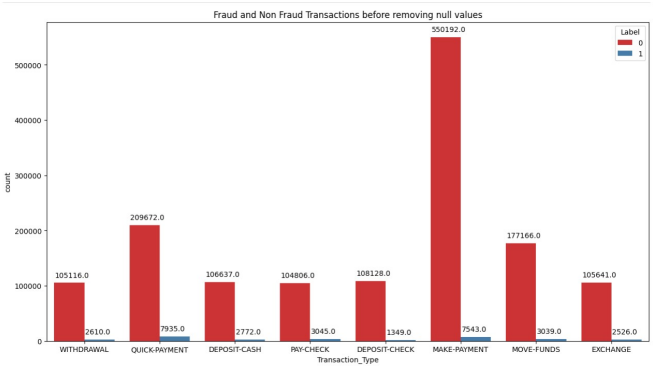


Fig. 1. Distribution of classes in the dataset

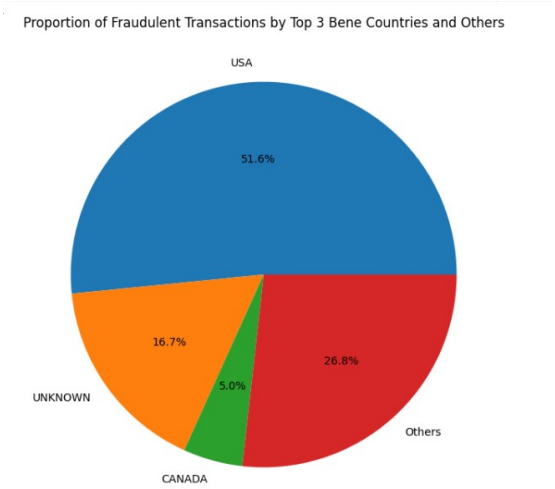
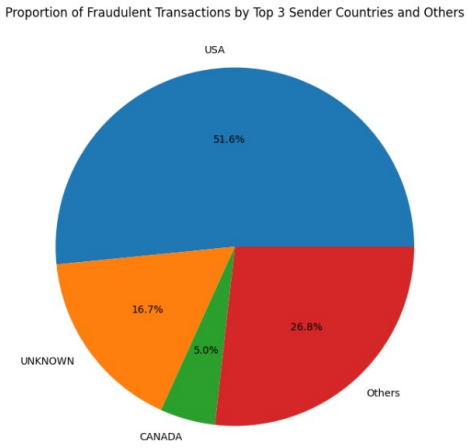


Fig. 2. Heat Map To Check The Correlation



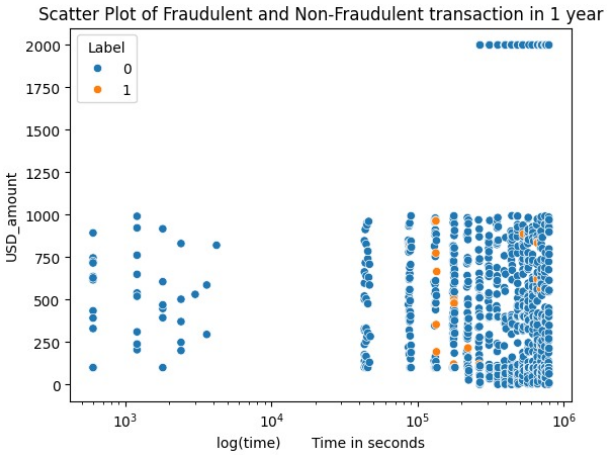


Fig. 3. Scatter Plot of the dataset

C. Dataset Visualizations - Paysim Dataset

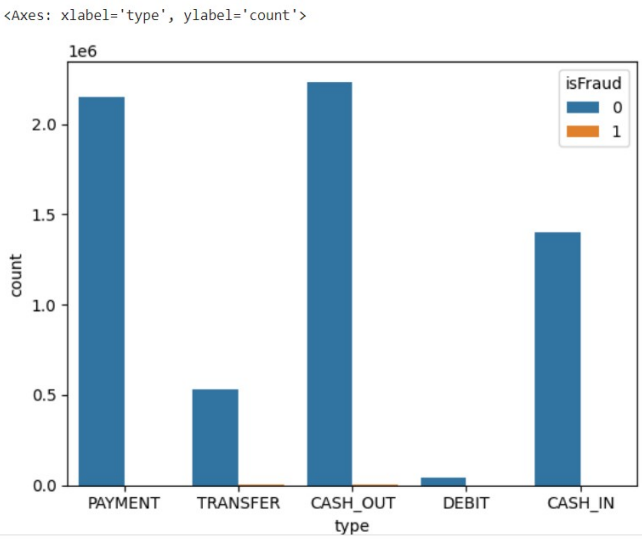


Fig. 4. Distribution of classes in the dataset

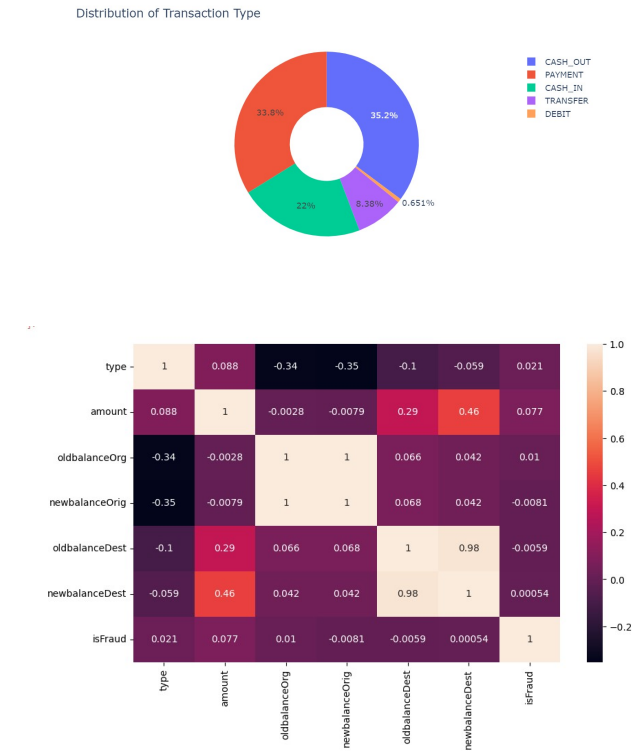


Fig. 5. Heat Map To Check The Correlation

IV. Methodology & Results Analysis

Dataset - JP Morgan

This section provides a detailed description of the methods used to build the classifier, discarded, and the results obtained during the study. Since, the data is a highly imbalanced data,

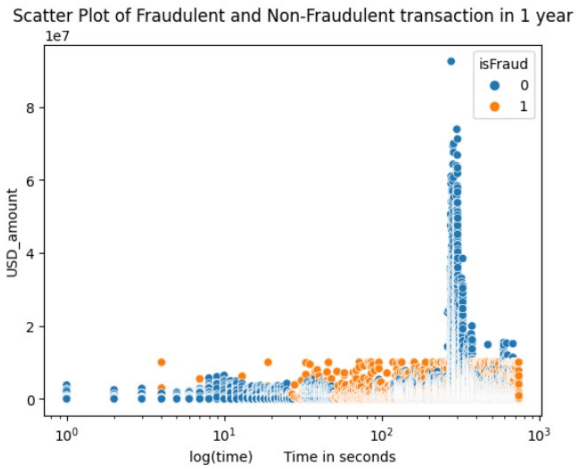


Fig. 6. Scatter Plot of the dataset

so the correct metrics to be used are precision, recall, F1-score and AUC-ROC. Accuracy is not a good measure to judge the model performance though it plays a vital role in judging False Positive cases. Since, no of true positive cases will ultimately affect False positive cases. We employed feature engineering on the dataset. There were majorly 5 parties between which transaction was caried - company, bill-company, client, JPMogan-client, JPMorgan-company. we employed feature engineering on Sender_Id and Bene_Id.Furthur, we employed feature engineering by creating a new column "surge inddicator". It was observed that fraud transaction have higher mean in the amount of tranfer than non-fruad. Surge indicator is a new column which has 1 if the transaction amount is greater than the threshold else it will be 0

1) Logistic Regression: A statistical method used to analyze and model between a (usually binary) dependent variable and one or more independent variables. This regression analysis technique aims to predict the probability of an unseen example belonging to a particular class. We discarded this approach because the dataset is highly imbalanced, meaning one class significantly outnumbers the other, the logistic regression model becomes biased towards the majority class and never predict a transaction as Fraud as depicted by the confusion matrix given below [fig. 8]. Also, the scatter plot of the data points from different classes overlaps significantly [fig. 3], the logistic regression works by fitting a linear decision boundary to separate classes. When classes overlap substantially in the feature space, a linear decision boundary might not be able to effectively distinguish between them, resulting in low predictive performance.

2) Anomaly Detection Algorithms: In data analysis, anomaly detection refers to the identification of observations which deviate significantly from the majority of the data and do not conform to a well-defined notion of normal behaviour. Such examples may arouse suspicions of being generated by a different mechanism, or appear inconsistent with the remainder of the dataset. These unsupervised learning models

	precision	recall	f1-score	support
Not Fraud	0.98	1.00	0.99	440208
Fraud	0.87	0.02	0.03	9246
accuracy			0.98	449454
macro avg	0.93	0.51	0.51	449454
weighted avg	0.98	0.98	0.97	449454

Fig. 7. Classification Report of Logistic Regression Model

are designed to identify patterns, relationships, or structures in data without using explicit labels or predefined outcomes. These models explore the data and find hidden patterns or groupings without the need for labeled responses.

The Isolation Forest procedure was used to detect outliers in the dataset.

Isolation Forest

It works by isolating observations in a dataset by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of that feature. This process is repeated recursively until the data points are isolated into individual trees, hence the name ”Isolation Forest.” Anomalies are then identified as instances that require fewer splits to isolate, indicating they are different from the majority of the data. The Data provided by the JP Morgan had many Fraudulent transaction which doesn't have anomalous nature. As a result, Isolation Forest gave a good recall, but a very poor precision and classified majority of the points as Fraud only.Isolation Forest started treating maximum points as anomalous which was absurd. Isolation Forest gave a very bad Accuracy Score.

Random Forest Classifier

It is an ensemble learning method that operates by constructing multiple decision trees during training and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. we build 100 trees and Random Forest gave us a very good results as the dataset was very complex and hard to interpret. Even employing aggresive feature engineering there were very few distinctive feature between fraudulent and legitimate transaction.

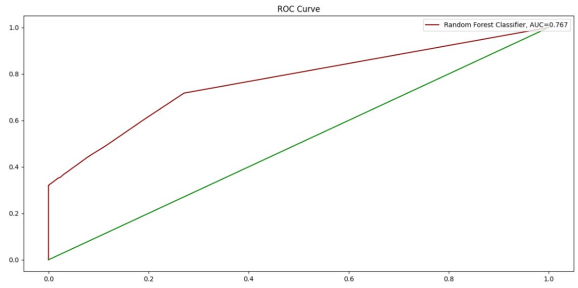


Fig. 8. ROC Curve of Random Forest

LGBM

It works by sequentially adding weak learners (decision trees) to an ensemble, with each subsequent learner focusing on the mistakes made by the previous ones. LightGBM optimizes the traditional gradient boosting algorithm by using a novel technique called Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). GOSS focuses on instances with large gradients during the training process, which helps in reducing the computational cost and memory usage. LGBM gave us good results on precision, recall, F1-score and exceptional AUC-ROC.

	precision	recall	f1-score	support
Not Fraud	0.99	1.00	0.99	440208
Fraud	1.00	0.32	0.48	9246
accuracy			0.99	449454
macro avg	0.99	0.66	0.74	449454
weighted avg	0.99	0.99	0.98	449454

col_0	0	1
Label		
0	440208	0
1	6287	2959

Fig. 9. Classification Report of LightGBM Model

XG Boost

XGBoost is based on the gradient boosting framework, which sequentially adds weak learners (decision trees) to an ensemble, with each tree trained to correct the errors of the previous ones. XGBoost also gave us a good precision but a low recall score. That is it was unable to classify more fraud transactions. The result from XG Boost was just satisfactory. Though it gave us a good AUC-ROC score.

Conclusion

For the given Dataset, the LGBM technique and random forest classifier gave the best result. Logistic gave the poor performance. The results are depicted in [fig. 10 & 11].

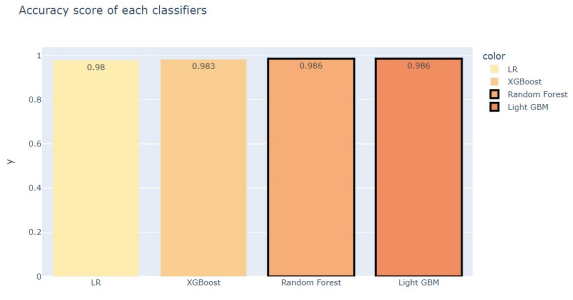


Fig. 10. Accuracy of the models

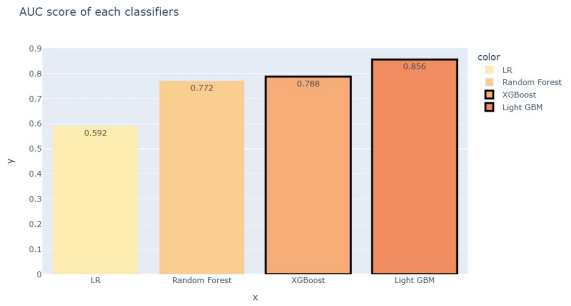


Fig. 11. AOC of models

Dataset - Paysim

3) Logistic Regression: A statistical method used to analyze and model between a (usually binary) dependent variable and one or more independent variables. We discarded this approach because of the same reason as mentioned above for the JP Morgan dataset, this dataset is highly imbalanced, meaning one class significantly outnumbers the other, the logistic regression model becomes biased towards the majority class and rarely predict a transaction as Fraud as depicted by the confusion matrix given below [fig. 12]. Also, the scatter plot of the data points from different classes overlaps significantly [fig. 6], the logistic regression works by fitting a linear decision boundary to separate classes. When classes overlap substantially in the feature space, a linear decision boundary might not be able to effectively distinguish between them, resulting in low predictive performance.

```
Logistic Regression Model:
[[635377     64]
 [   467    354]]
Accuracy score on Test Data : 0.9991654381371197%
              precision    recall  f1-score   support

Non-Fraud [0]       1.00      1.00      1.00    635441
    Fraud [1]       0.85      0.43      0.57       821

     accuracy                           1.00    636262
    macro avg       0.92      0.72      0.79    636262
 weighted avg       1.00      1.00      1.00    636262
```

Fig. 12. Classification Report of Logistic Regression Model

4) Classifiers: Various classifiers including KNeighbours Classifier, Random Forest Classifier, Decision Tree Classifier, and XGBoost were evaluated for fraud detection. Decision Tree and Random Forest emerged as the top-performing classifiers based on metrics such as F1 score and precision. The results are depicted in [fig. 14 & 15].

However, due to the highly imbalanced nature of the dataset, where the number of non-fraudulent transactions far exceeded the number of fraudulent transactions, additional techniques were explored to address this imbalance :-

1. Undersampling: Attempted to balance the class distribution by reducing the number of instances in the majority class.

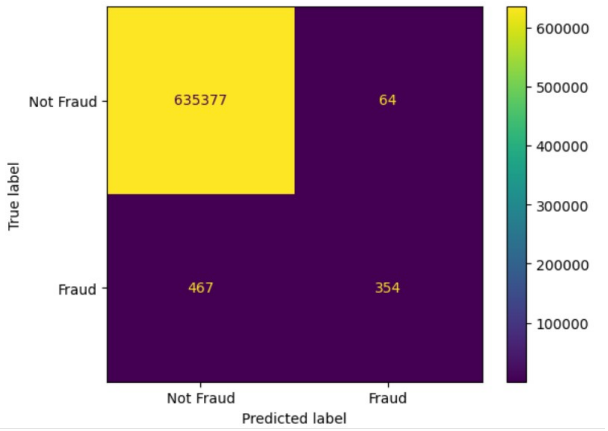


Fig. 13. Confusion Matrix of Logistic Regression Model

```
Decision Tree Classifier Model:
[[635373     68]
 [    83    738]]
Accuracy score on Test Data : 0.9997626763817421%
              precision    recall  f1-score   support

Non-Fraud [0]       1.00      1.00      1.00    635441
    Fraud [1]       0.92      0.90      0.91       821

     accuracy                           1.00    636262
    macro avg       0.96      0.95      0.95    636262
 weighted avg       1.00      1.00      1.00    636262
```

Fig. 14. Classification Report of Decision Tree Classifier

However, by significantly reducing the number of instances in the majority class, led to a drastic decrease in the representation of non-fraudulent transactions. This loss of crucial data points hindered the model's ability to generalize effectively, resulting in a skewed understanding of the majority class and ultimately impacting its performance in accurately identifying non-fraudulent transactions. The results are depicted in [fig. 16].

2. Oversampling: Augmented the minority class by

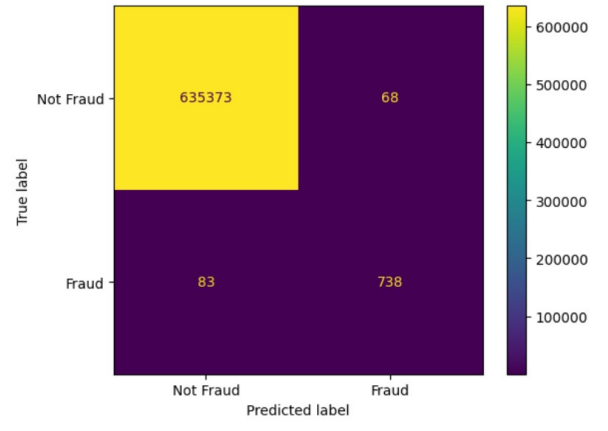


Fig. 15. Confusion Matrix of Decision Tree Model

```
Random Forest Model:
[[628524   6917]
 [     2    819]]
Accuracy score on Test Data : 0.9891255489090972%
              precision    recall  f1-score   support

Non-Fraud [0]       1.00      0.99      0.99    635441
    Fraud [1]       0.11      1.00      0.19       821

     accuracy                           0.99    636262
    macro avg       0.55      0.99      0.59    636262
 weighted avg       1.00      0.99      0.99    636262
```

Fig. 16. Classification Report of Random Forest with Undersampled dataset

duplicating instances or generating synthetic samples. While it did not significantly improve performance over the unbalanced dataset, it did not exacerbate the issues seen with undersampling. 3. SMOTE (Synthetic Minority Over-sampling Technique): Generated synthetic samples for the minority class by interpolating between existing instances. Like oversampling, the resulting models performed similarly to those trained on the unbalanced dataset.

Conclusion

This suggests that the original classifiers, particularly Decision Tree and Random Forest, were robust enough to handle the class imbalance without the need for additional sampling techniques, highlighting their effectiveness in capturing the underlying patterns within the data. The results are depicted in [fig. 17].

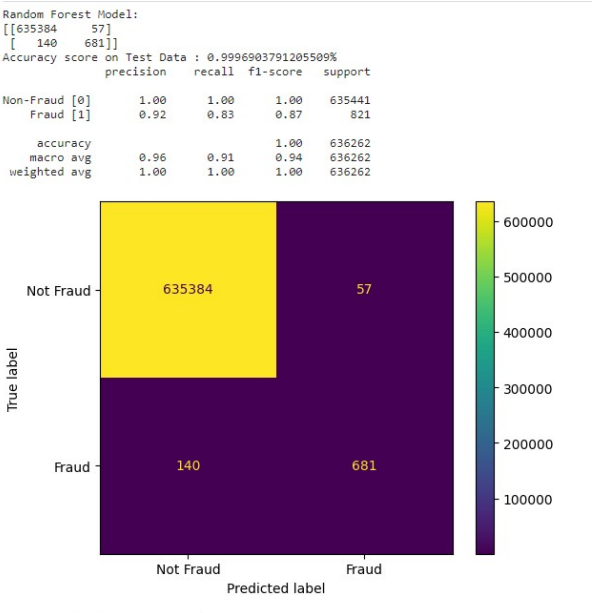


Fig. 17. Classification Report of Random Forest with Oversampled dataset

REFERENCES

[1] Siddhartha Bhattacharyya, Sanjeev Jha, Kurian Tharakunnel, J. Christopher Westland, Data mining for credit card fraud: A comparative study, Decision Support Systems, v.50 n.3, p.602-613, February 2011

[2] Maniraj, S & Saini, Aditya & Ahmed, Shadab & Sarkar, Swarna. (2019). Credit Card Fraud Detection using Machine Learning and Data Science. International Journal of Engineering Research and. 08. 10.17577/IJERTV8IS090031.

[3] Research on Credit Card Fraud Detection Model Based on Distance Sum – by Wen-Fang YU and Na Wang" published by the 2009 International Joint Conference on Artificial Intelligence.