Financial Fraud Detection

Milestone: Project proposal, Data collection and processing, etc.

Group 1
Deepak Kumar Meena

+1 857 415 1106

meena.d@northeastern.edu

Percentage of Effort Contributed by Student 1: 100

Signature of Student 1: Deepak Kumar Meena

Submission Date: 05/03/2022

PROBLEM SETTING:

Digital payments in various forms are on the increase all over the world. The transaction volume handled by financial institutions is rapidly increasing. In 2018, PayPal, for example, processed $578 billion in total payments. Along with this change, there has been a significant surge in financial fraud in various payment systems.
The role of cybersecurity and cyber-crime teams includes preventing online financial fraud. Most banks and financial organizations have specialized teams of dozens of analysts working on automated systems to monitor transactions made through their products and flag those that are possibly fraudulent. As a result, in order to be better prepared to address the challenge of identifying fraudulent entries/transactions in vast volumes of data, it is critical to investigate the strategy to tackling the problem.

PROBLEM DEFINITION:

Payments-related fraud is a major concern for cyber-crime organizations, and recent research has demonstrated that machine learning approaches may successfully detect fraudulent transactions in vast amounts of data. Such algorithms can detect fraudulent transactions that human auditors may be unable to identify, and they can do so in real time.
Using publicly available simulated payment transaction data, we apply different supervised machine learning algorithms to the problem of fraud detection in this research. We want to show how supervised machine learning techniques may be utilized to accurately categorize data with substantial class imbalance.
We show how to use exploratory analysis to distinguish between fraudulent and non-fraudulent transactions. In addition, we show that a well separated dataset, tree- based algorithms like Random Forest and Decision tree work much better than Logistic Regression.

DATA SOURCE:

The dataset was obtained from Kaggle which describes transactions which collected data of over 6,353,307 transactions. It presents a synthetic dataset generated using the simulator called PaySim.

Link: https://www.kaggle.com/datasets/ealaxi/paysim1

DATA DESCRIPTION:

The dataset consists of 6.3M rows and 11 columns. Columns involve step (maps a unit of time in the real world. In this case 1 step is 1 hour of time. Total steps 744 i.e., 30 days simulation), type (CASH-IN, CASH-OUT, DEBIT, PAYMENT and TRANSFER), amount (amount of the transaction in local currency), nameOrig (customer who started the transaction), oldbalanceOrg (initial balance before the transaction), newbalanceOrig (new balance after the transaction) nameDest (customer who is the recipient of the transaction), oldbalanceDest (initial balance recipient before the transaction), newbalanceDest (new balance recipient after the transaction), isFraud (In this specific dataset the fraudulent behavior of the agents aims to profit by taking control or customers' accounts and try to empty the funds by transferring to another account and then cashing out of the system) and

isFlaggedFraud (An illegal attempt in this dataset is an attempt to transfer more than 200.000 in a single transaction)

DATA EXPLORATION:

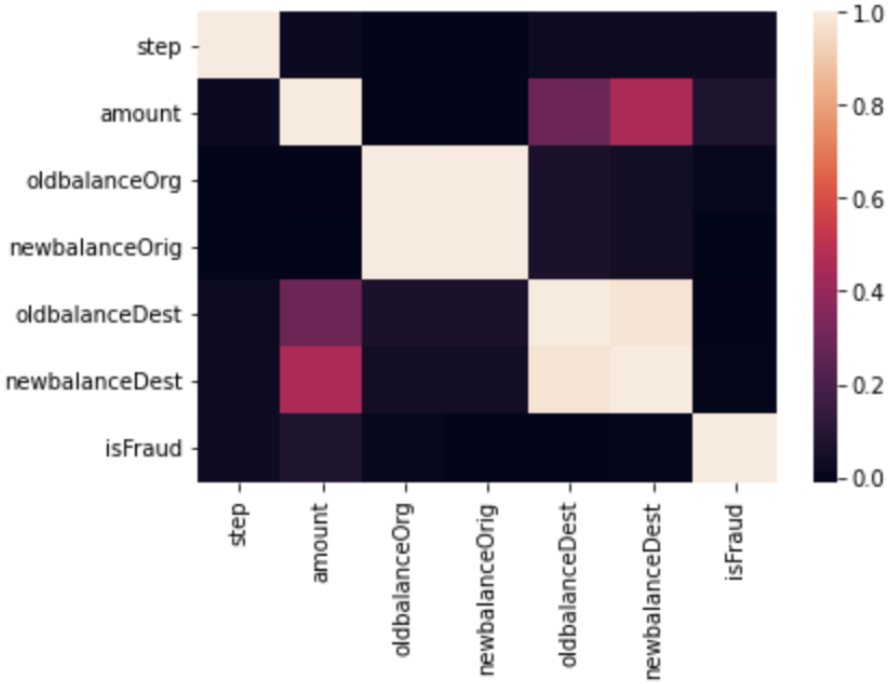
Dataset:

	step	type	amount	nameOrig	oldbalanceOrg	newbalanceOrig	nameDest	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud
0	1	PAYMENT	9839.64	C1231006815	170136.0	160296.36	M1979787155	0.0	0.0	0	0
1	1	PAYMENT	1864.28	C1666544295	21249.0	19384.72	M2044282225	0.0	0.0	0	0
2	1	TRANSFER	181.00	C1305486145	181.0	0.00	C553264065	0.0	0.0	1	0
3	1	CASH_OUT	181.00	C840083671	181.0	0.00	C38997010	21182.0	0.0	1	0
4	1	PAYMENT	11668.14	C2048537720	41554.0	29885.86	M1230701703	0.0	0.0	0	0

Check for na values:

```
step              0
type              0
amount            0
nameOrig          0
oldbalanceOrg     0
newbalanceOrig    0
nameDest          0
oldbalanceDest    0
newbalanceDest    0
isFraud           0
isFlaggedFraud    0
dtype: int64
```

Correlation plot between numerical variables before data reduction and data transformation:



As there is no significant correlation between the numerical variables.

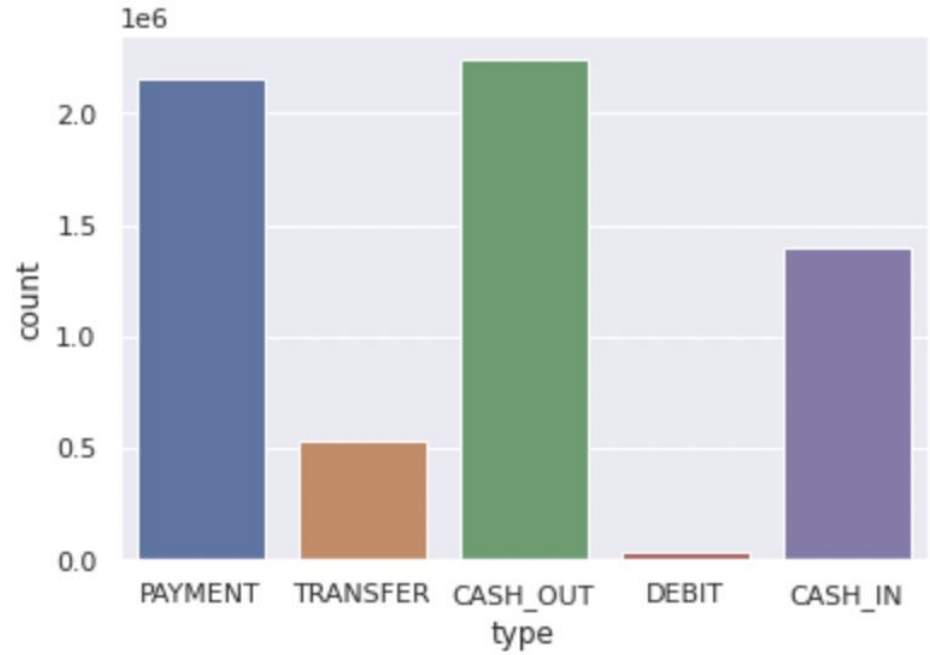
Count plot for isFraud:

It shows that 99.87% of the transactions are not fraud and only 0.13% of the transactions are fraud.

Count plot for isFlaggedFraud:

It shows that 99.99% of the transactions are not fraud and fraud transactions are almost negligible. Hence, we can drop it as it won't make any difference.

Transactions according to the type:



CASH_OUT and PAYMENT are the major transaction types. CASH_OUT(35.16%), PAYMENT(33.81%), CASH_IN(21.99%), TRANSFER(8.37%) and DEBIT(0.65%).

Percentage according to isFraud:

```
isFraud  type
0        CASH_OUT    35.101641
         PAYMENT     33.814608
         CASH_IN     21.992261
         TRANSFER     8.311230
         DEBIT        0.651178
1        CASH_OUT     0.064690
         TRANSFER     0.064392
Name: type, dtype: float64
```

Only CASH_OUT and TRANSFER are the types were fraud transactions. Hence, we can select the data only with these 2 types of transactions.

Now the updated dataset has 2,770,409 after dropping isFlaggedFraud and filtering by CASH_OUT and PAYMENT type of transactions.

Sanity check:

Check for the negative or zero amount because transactions cannot be negative or zero. 16 transactions came out to be zero, so we drop them as well.

Now, we create two new variables with inaccuracy in the origin and destination transactions to get more overview and correlation with the fraud transactions.

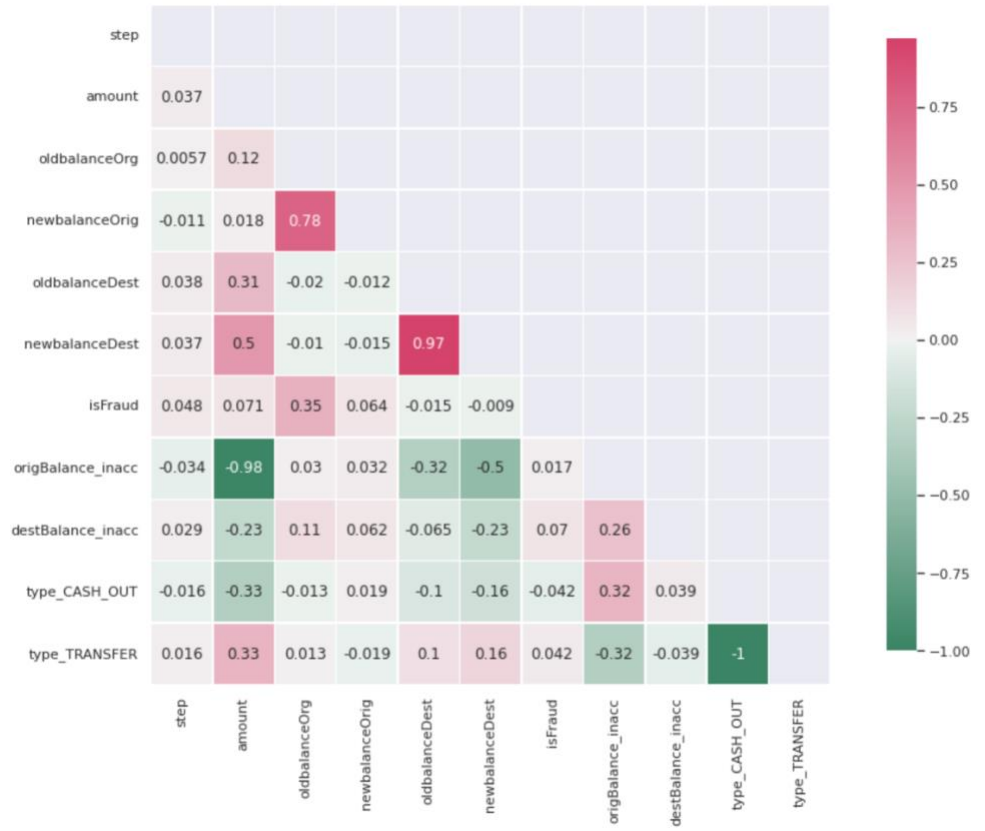
```
fraud['origBalance_inacc'] = (fraud['oldbalanceOrg'] - fraud['amount']) -
fraud['newbalanceOrig']
fraud['destBalance_inacc'] = (fraud['oldbalanceDest'] + fraud['amount'])
- fraud['newbalanceDest']
```

Now, we want only the numerical data hence drop all the names. Create dummies for the type of transaction to convert it to categorical type.

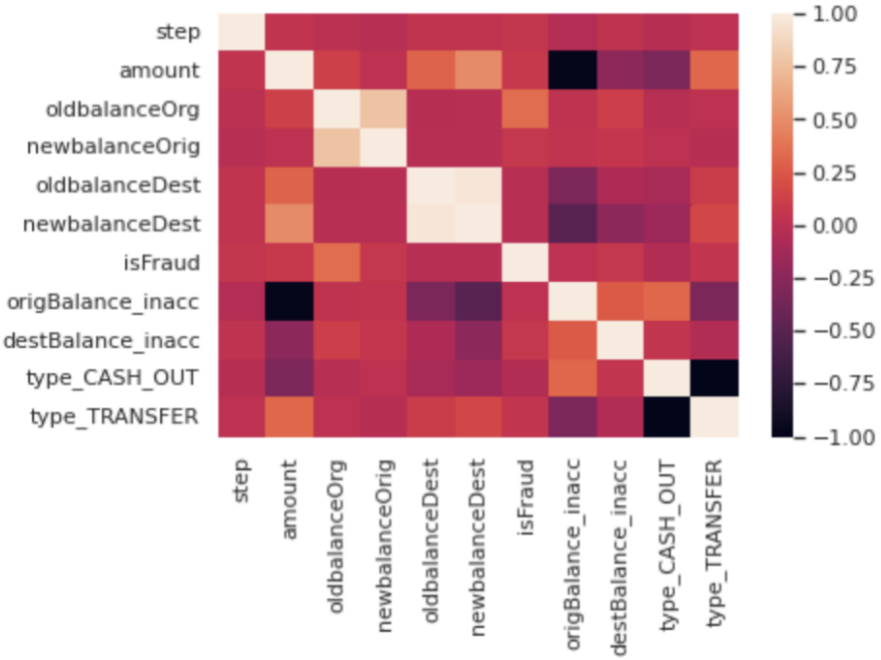
Updated data:

	step	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDest	isFraud	origBalance_inacc	destBalance_inacc	type_CASH_OUT	type_TRANSFER
2	1	181.00	181.0	0.0	0.0	0.00	1	0.00	181.0	0	1
3	1	181.00	181.0	0.0	21182.0	0.00	1	0.00	21363.0	1	0
15	1	229133.94	15325.0	0.0	5083.0	51513.44	0	-213808.94	182703.5	1	0
19	1	215310.30	705.0	0.0	22425.0	0.00	0	-214605.30	237735.3	0	1
24	1	311685.89	10835.0	0.0	6267.0	2719172.89	0	-300850.89	-2401220.0	0	1

Correlation plot after reduction, filtering and creating new variables:



There is 97% correlation between ‘old destination balance’ and ‘new destination balance’ along with 78% correlation between ‘old origin balance’ and ‘new origin balance’.



DATA MINING TASK:

Dimension Reduction:

Column isFlaggedFraud has almost negligible fraudulent therefore, we drop that column. type column has 5 types of payment of which only 2 types have fraudulent values. Therefore, we select the values only with type CASH_OUT and TRANSFER. So, they are not useful for prediction.

Categorical Variables:

Converted the modified type column to categorical variables using pd.get_dummies() method and formed new columns for different types with binary values.

New variables:

Created new variables which represent the inaccuracy in the initial and final balance in the account by keeping the amount into consideration.

DATA PARTITIONING:

The required columns are all stored in the variable X and the target column was stored in Y. Then they are divided into 70% train and 30% test namely X_train, X_test, y_train, y_test. X_train had1939275 rows & 10 columns and X_test had 831118 rows & 10 columns. y_train and y_test had the same number of rows as X_train and X_test respectively.

DATA MINING MODELS/METHODS:

This is a classification problem in which we must predict the amount of satisfaction based on the data provided. After doing the initial data wrangling and selecting the required featured that will be useful in the prediction, we splitted the dataset into training (70%) and validation (30%) data. Following that, we implemented the classification methods shown below, choosing the one that delivered the best classification metrics.

Decision tree classifier:

Decision Tree Classifier uses a decision tree (as a predictive model) to go from observations about an item to conclusions about the item's target value (represented in the leaves). Tree models where the target variable can take a discrete set of values are called classification trees.

Advantages:

1. Simple to understand and interpret
2. Able to handle both numerical and categorical data
3. Possible to validate a model using statistical tests
4. Performs well with large datasets

Disadvantages:

1. Trees can be very non-robust
2. Decision-tree learners can create over-complex trees that do not generalize well from the training data
3. The average depth of the tree that is defined by the number of nodes or tests till classification is not guaranteed to be minimal

Random forest classifier:

Random forest, like its name implies, consists of many individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction.

Advantages:

1. Compared to other algorithms decision trees requires less effort for data preparation during pre-processing
2. A decision tree does not require normalization of data.
3. A decision tree does not require scaling of data as well.
4. Missing values in the data also do NOT affect the process of building a decision tree to

   any considerable extent.

Disadvantages:

1. A small change in the data can cause a large change in the structure of the decision tree causing instability.
2. For a Decision tree sometimes, calculation can go far more complex compared to other algorithm

Logistic Regression:

Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval, or ratio-level independent variables.

Advantages:

1. Simplest machine learning algorithm
2. Easy to update
3. Well-calibrated outputs
4. Less prone to over-fitting
5. More accurate

Disadvantages:

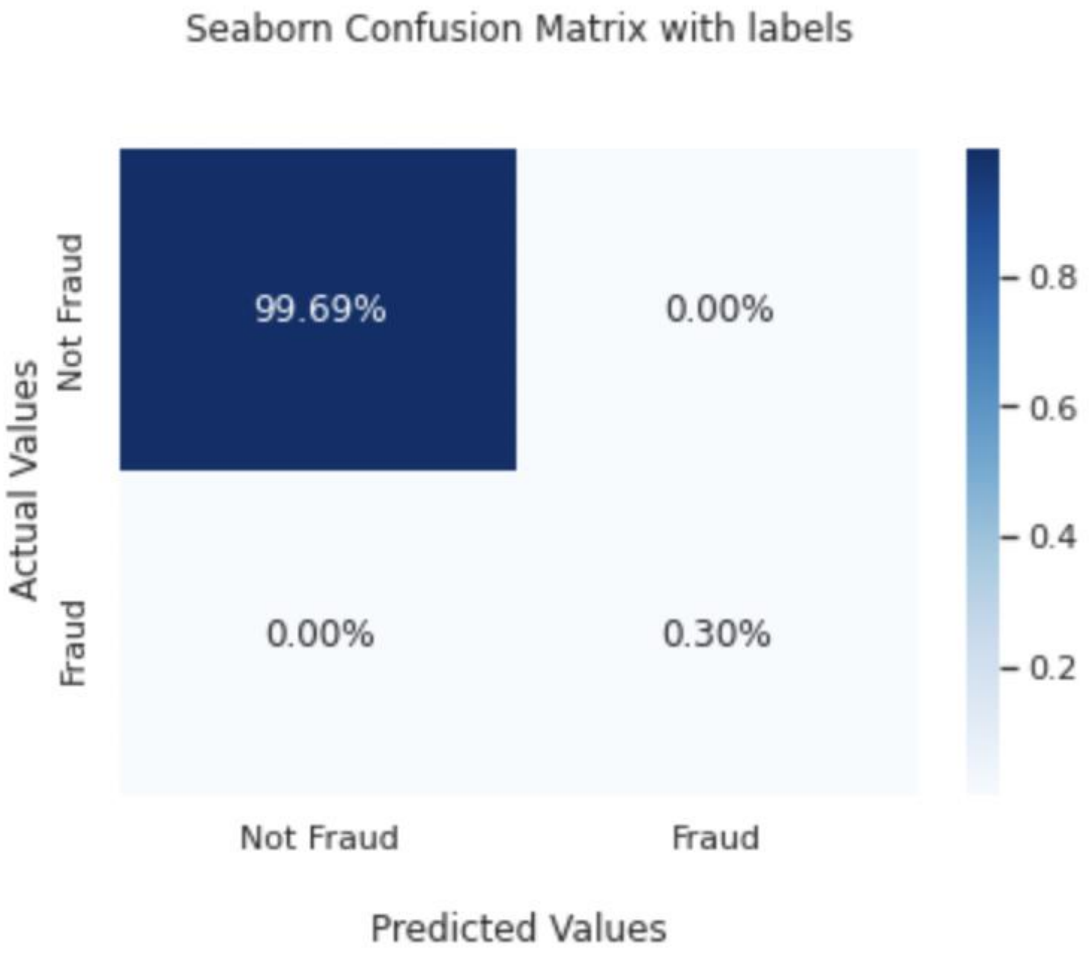
1. Over-fitting
2. Not all problems are solvable using this approach
3. Problem with complex relationships
4. Requires a high number of observations

PERFORMANCE EVALUATION:

Decision tree classifier:

A decision tree classifier was built to train the data using the default parameters and the model's performance is as recorded below.

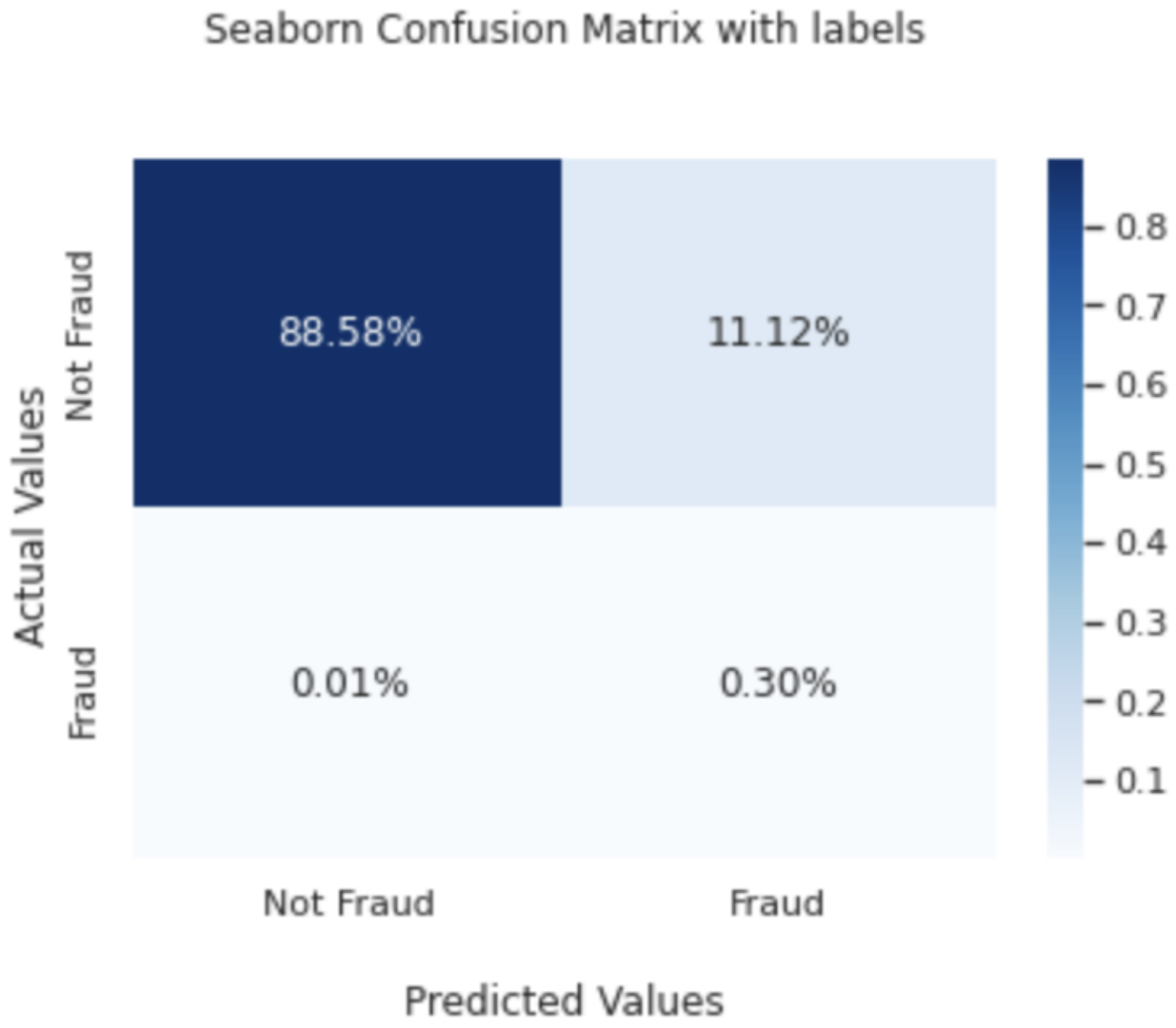
```
Accuracy Score: 0.9999795456240871
F1 Score: 0.9966528844260681
Precision Score: 0.9976350019708317
Recall Score: 0.9956726986624705
```



Logistic Regression:

A logistic regressor was built to train the data using the default parameters and the model's performance is as recorded below.

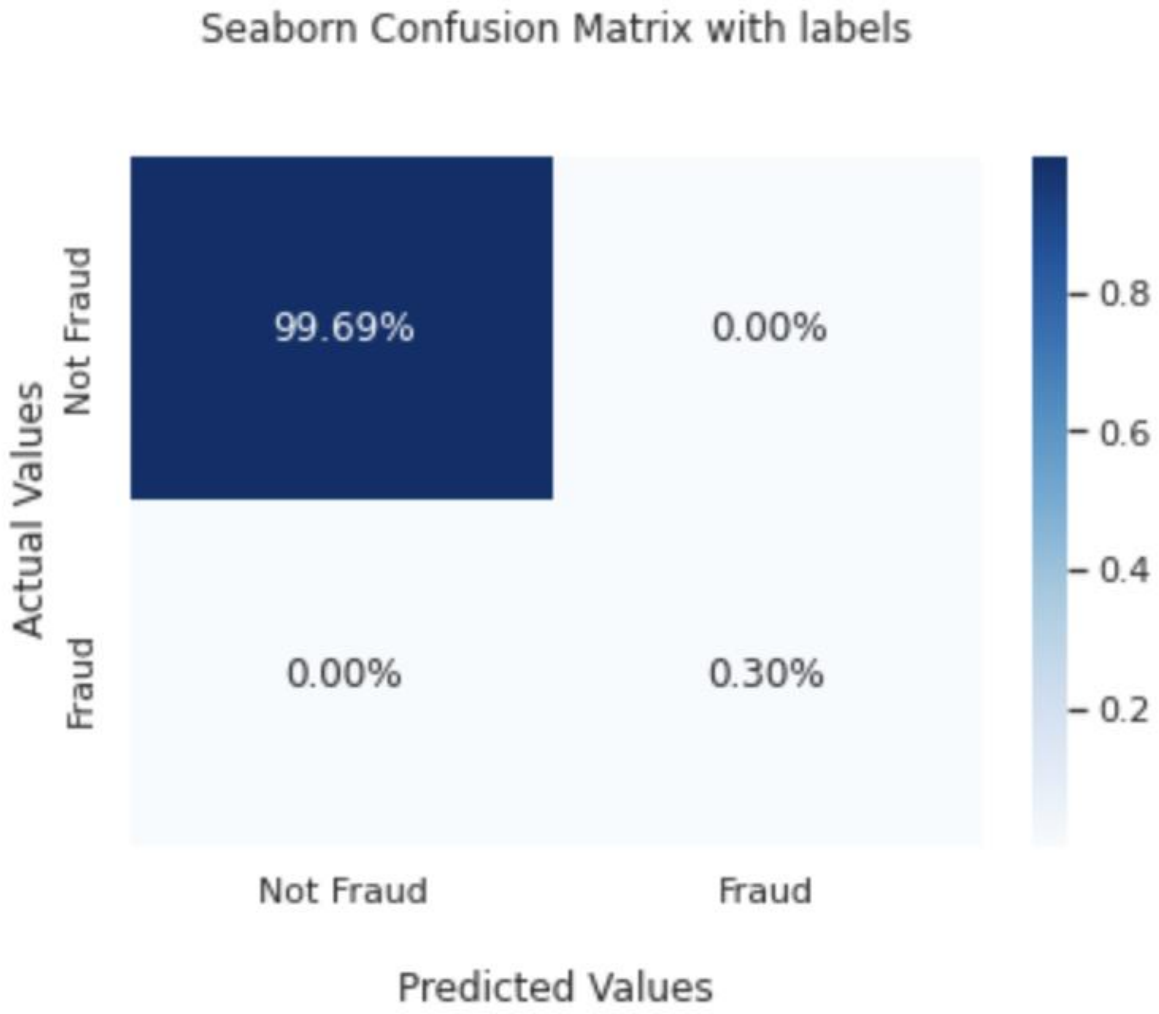
```
Accuracy Score: 0.8887570717996722
F1 Score: 0.05075975359342916
Precision Score: 0.026060005481878177
Recall Score: 0.9724626278520849
```



Random forest classifier:

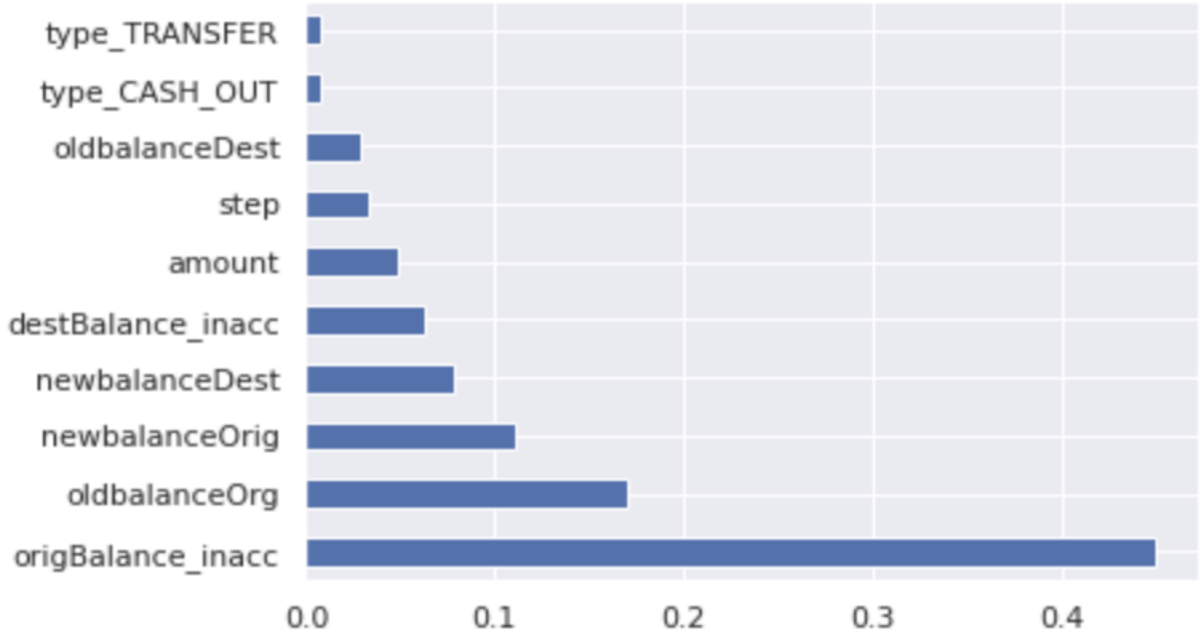
A random forest classifier was built with the default parameters and the below results were obtained.

```
Accuracy Score: 0.9999867648155858
F1 Score: 0.997832512315271
Precision Score: 0.999605211212001 6
Recall Score: 0.996066089693155
```



PROJECT RESULTS:

Random forest classifier was chosen to predict the fraudulent transaction and the factors which mainly affect are given below.



Origin Balance inaccuracy, the created variable is the most important factor which affects the fraudulent transaction followed old balance of origin.

The other important factors in order are origin new balance, destination new balance and destination balance inaccuracy.

So, the focus should be more on these conditions to recognize fraudulent transactions.