Fraud Detection in Mobile Money Transactions

The Problem

The aim of this project is to use computer simulation for fraud detection in mobile money financial transactions. Fraud prevention is becoming a critical driver for the financial services industry. In the recent years, due to an increase in use of new technologies such as cloud and mobile financial services, the fraud problem has been intensified. Therefore achieving an accurate and less intrusive fraud detection system is crucial and financial services institutions are increasingly investing in algorithms and data analytics technology to spot and combat fraud.

Clients

The primary client for this project would be banks and financial services that provide mobile money transactions. However, financial fraud is a problem which affects the finance industry, government, corporate sectors, and ordinary consumers, and therefore identifying and preventing fraud can be beneficial for many different clients.

Current Dataset

Currently, there is a lack of public research into the detection of fraud. One important reason is shortage of transaction data due to confidentiality issues. Due to this problem, a synthetic dataset generated using the simulator called PaySim is used in this project. PaySim uses aggregated data from the private dataset to generate a synthetic dataset that resembles the normal operation of transactions and injects malicious behaviour to later evaluate the performance of fraud detection methods. The data is described in details in the following PhD thesis: http://urn.kb.se/resolve?urn=urn:nbn:se:bth-12932

Description:

The Paysim synthetic dataset of mobile money transactions available on Kaggle is used for this project. The transaction data is presented in different steps, each step representing an hour of simulation. The raw data consists of 11 columns and about 6362620 rows. The description of each column is as follows:

step: Maps a unit of time in the real world. In this case, one step is one hour of time. Total steps are 744 (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.
oldbalanceOrig: 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. Note that there is no information for customers that start with M (Merchants).
newbalanceDest: new balance recipient after the transaction. Note that there is no information for customers that start with M (Merchants).
isFraud: This is the transactions made by the fraudulent agents inside the simulation. In this specific dataset the fraudulent behaviour 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.
isFlaggedFraud: The business model aims to control massive transfers from one account to another and flags illegal attempts. An illegal attempt in this dataset is an attempt to transfer more than 200.000 in a single transaction.

Transaction types are defined based on the following reference: http://bth.diva-portal.org/smash/get/diva2:955852/FULLTEXT06.pdf
CASH-IN is the process of increasing the balance of account by paying in cash to a merchant.
CASH-OUT is the opposite process of CASH-IN, it means to withdraw cash from a merchant which decreases the balance of the account.
DEBIT is similar process than CASH-OUT and involves sending the money from the mobile money service to a bank account.
PAYMENT is the process of paying for goods or services to merchants which decreases the balance of the account and increases the balance of the receiver.

TRANSFER is the process of sending money to another user of the service through the mobile money platform.

Data Wrangling

By looking at the legitimate versus fraudulent transactions, it can be seen that the number of fraudulent transactions are much lower than the legitimate ones. Therefore the data is highly imbalanced.

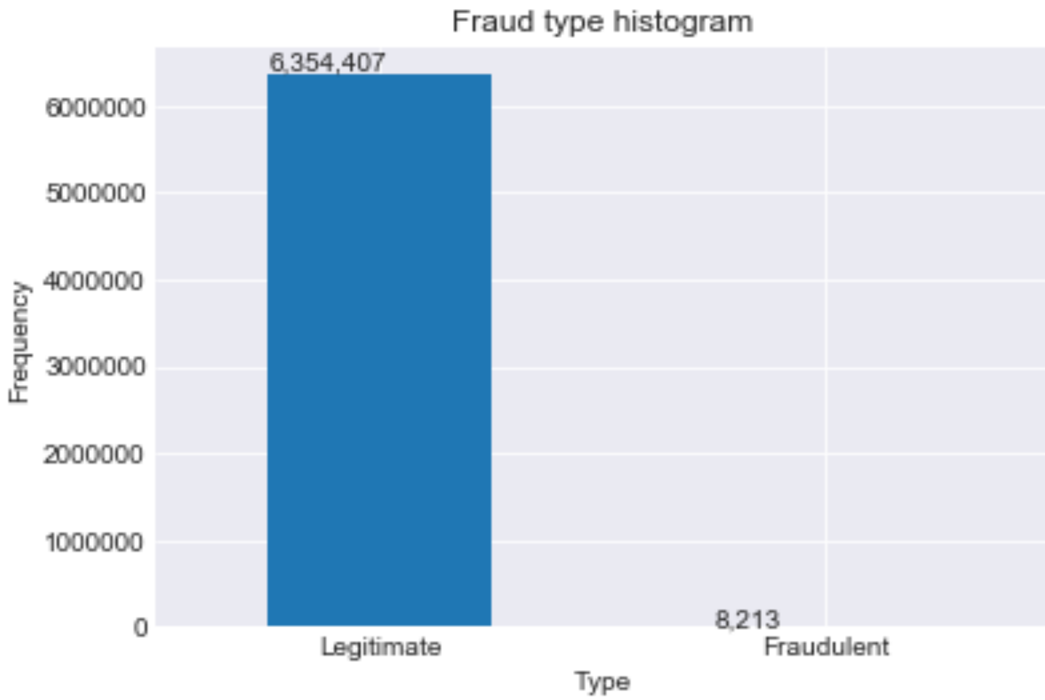
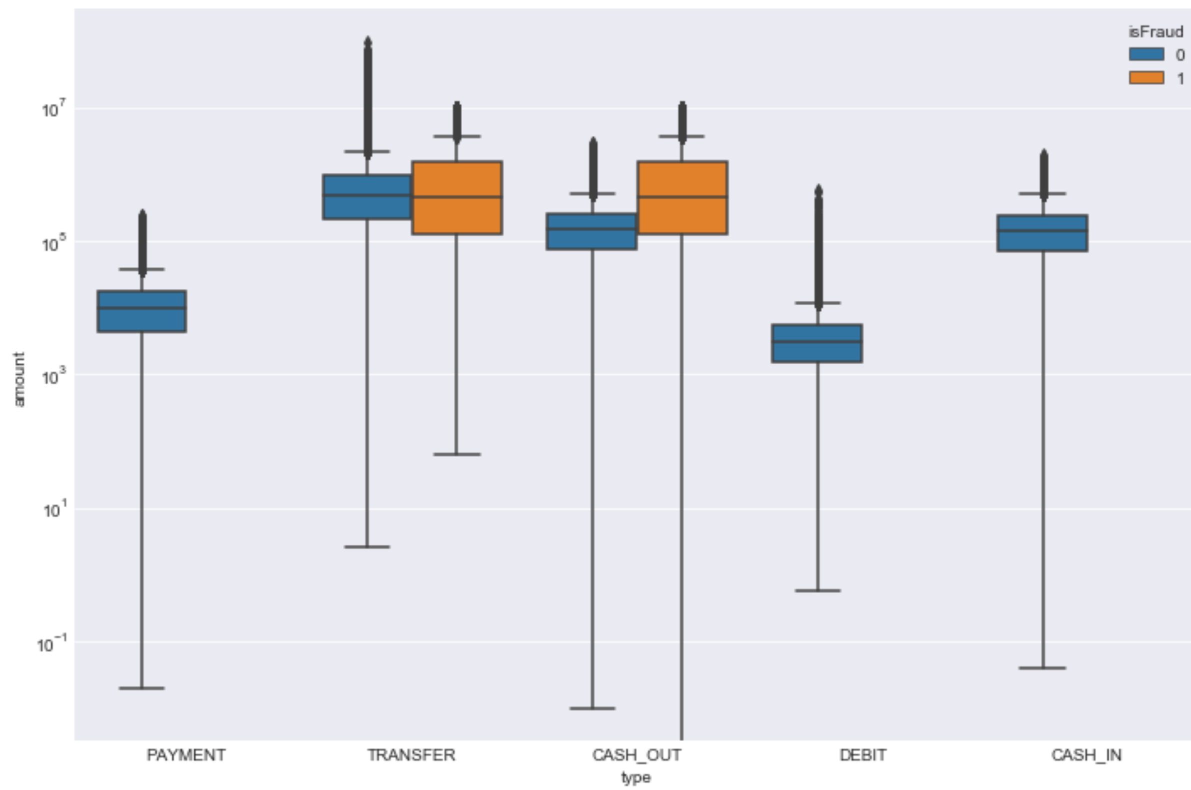


Figure below shows the pair plot for different features for both legitimate and fraudulent transactions.

Next we look at the amount variations for fraudulent and legitimate transactions. It can be seen that fraud only happens in TRANSFER and CASH-OUT transactions:

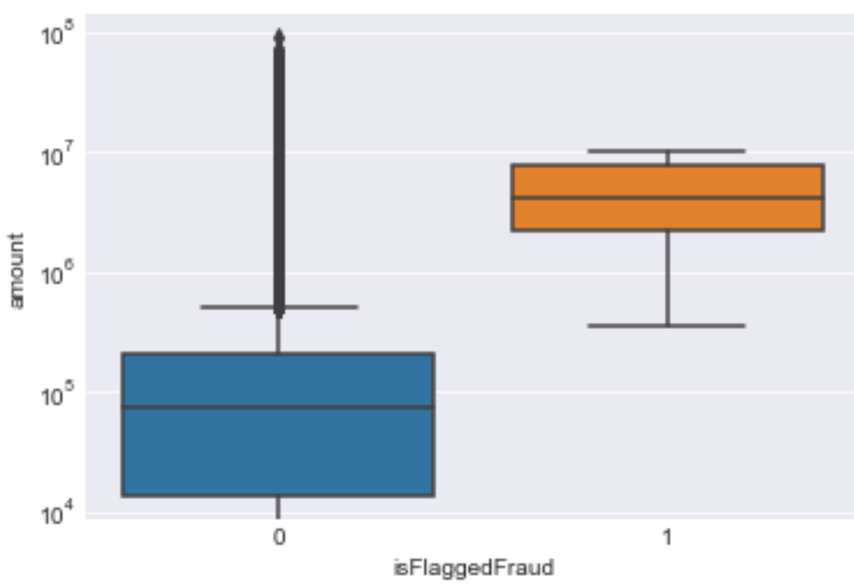
Number of TRANSFER fraudulent activities: 4097
Number of CASH-OUT fraudulent activities: 4116



It can also be seen that the amounts are higher for fraudulent transactions, in both TRANSFER and CASH-OUT.

Checking IsFlaggedFraud Feature

Next we look at IsFlaggedFraud feature. The amount for IsFlaggedFraud values of 0 and 1 is shown below. The amount for IsFlaggedFraud is higher than non Flagged amounts. Also the type of transactions in which isFlaggedFraud=1 is only TRANSFER.



As described in the features definitions, all the transactions which are "isFlaggedFraud" are the ones with transaction amounts higher than 200,000. We check to see if this holds for our data:

Number of isFlaggedFraud=1 cases where the amount if lower or equal to 200,000: 0
Number of TRANSFERs over 200,000 which are not flagged as isFlaggedFraud: 409094

So it seems that any TRANSFER over 200,000 is not necessarily flagged as isFlaggedFraud=1, which is in contrast with the definition.

Now we look at the correlation between isFraud and isFlaggedFraud:
Number of cases where isFlaggedFraud=1 but isFraud=0: 0
Number of cases where isFraud=1 but isFlaggedFraud=0: 8197

While whenever isFlaggedFraud=1, there is actually a fraud case, there are 8197 cases which are Fraud cases but isFlaggedFraud has failed to flag them as fraudulent. There are also only 16 TRANSFERs which are flagged as fraud.

Next we check the values and differences of oldBalanceDest and newBalanceDest for isFlaggedFraud and TRANSFER:

Sum of difference between new and old balance of destinations for flagged transfers: 0
Number of non-zero old balance of destinations for flagged transfers: 0

So any TRANSFER that has been flagged fraudulent, does not have any old or new destination balances. This can be due to suspension of the transfers because of the fraud flag. So destination balances do not give us any condition for flagging a transaction fraud.

By checking the number of times a customer with isFlaggedFraud TRANSFER had more than one transaction, it is observed that no customer with

FlaggedFraud TRANSFER had more than one transaction. Also there is no case in which the origin and destination of a FlaggedFraud TRANSFER is the same person.

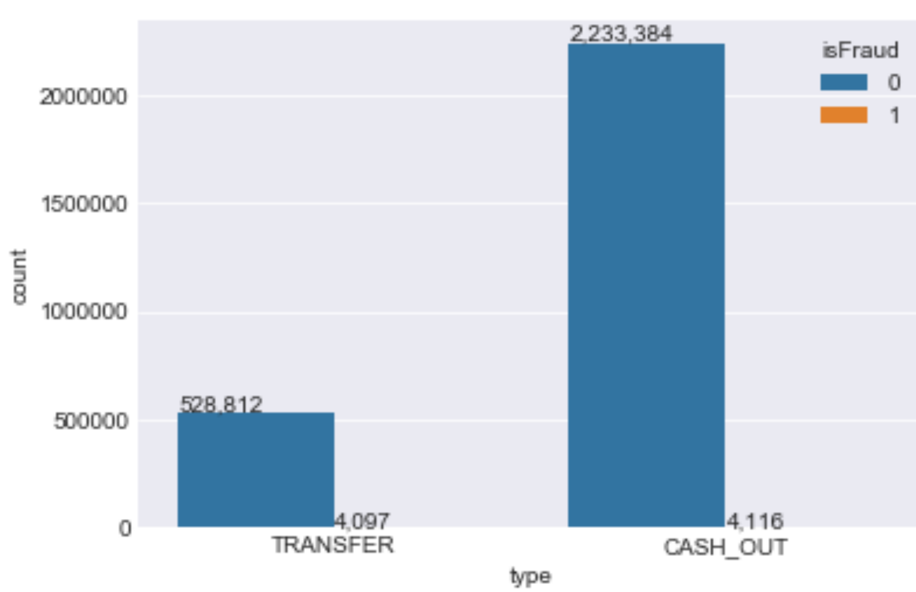
Looking at all the analysis above, it seems that isFlaggedFraud does not play any significant meaningful role in setting Fraud cases. Even though isFraud is set whenever isFlaggedFraud is set to true, isFlaggedFraud is on only for 16 TRANSFERs. Therefore, isFlaggedFraud column will be removed in this dataset.

Looking at NameDest feature

According to the definition of features mentioned above, a merchant should be involved in both CASH-IN and CASH-OUT transactions. However, by looking at the number of fraudulent cases involving merchants, one can realize that there is no record of balance from clients that start with M (Merchants). Since there are many CASH-OUT cases in fraudulent transactions, and we don't see any merchant involved in them, it can be concluded that the name columns are not defined properly and can be dropped.

Dropping other transaction types

Since there are only two types of transactions involved in a fraud (TRANSFER and CASH-OUT), we will delete the other rows.



Treating zero balances

There are lots of zero values in new and old destination balances even though the amount is not zero. These values may need to be replaced.

First we will take a look at the percentage of above condition for both fraudulent and legitimate transactions:
Percentage of fraudulent transactions where new and old destination balances are zero, even though the amount is not zero: 49.55 %
Percentage of legitimate transactions where new and old destination balances are zero, even though the amount is not zero: 0.06 %

Percentage of fraudulent transactions where new and old original balances are zero, even though the amount is not zero: 0.30 %
Percentage of legitimate transactions where new and old original balances are zero, even though the amount is not zero: 47.37 %

It is clear that the old and new destination balances being zero when amount is not zero happens much more often for fraudulent transactions. So it can be an indicator of the fraud. Therefore these values should not be replaced.

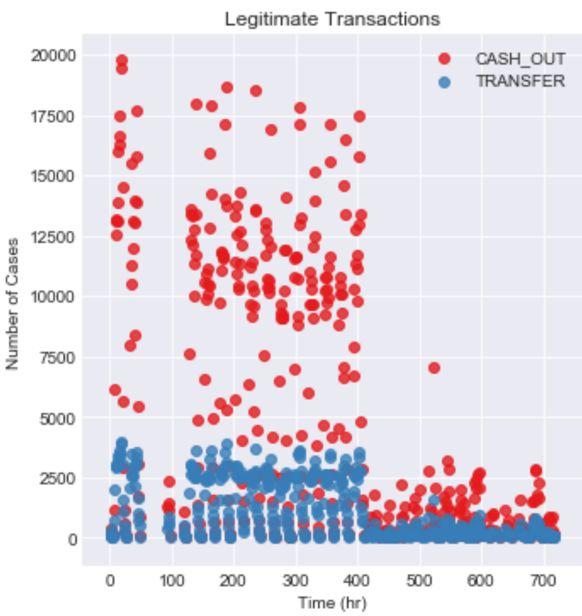
Also for original accounts, fraudulent transactions have much less percentage of new and old original balances being zero compared to legitimate transactions. For the same reason as above, we will not replace these numbers. Instead, we will create two new features calculating the error for both original and destination balances as follows:

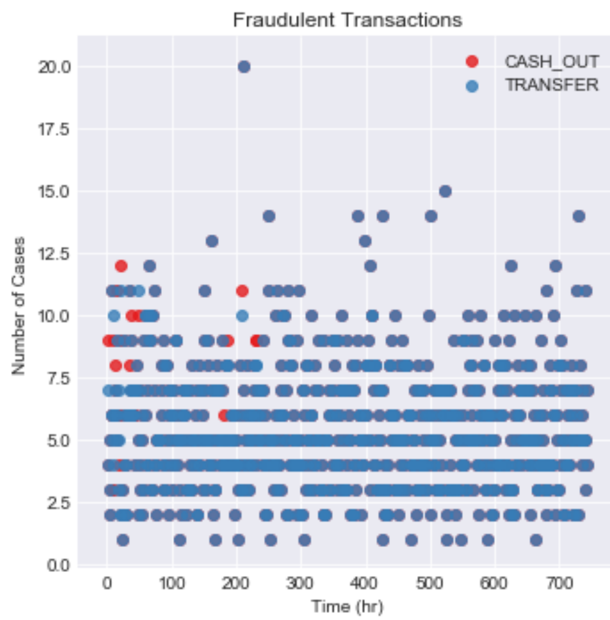
```
X_Fraud['errorBalanceOrig']=X_Fraud['newBalanceOrig']+X_Fraud['amount']-X_Fraud['oldBalanceOrig']
X_Leg['errorBalanceOrig']=X_Leg['newBalanceOrig']+X_Leg['amount']-X_Leg['oldBalanceOrig']
X_Fraud['errorBalanceDest']=X_Fraud['newBalanceDest']+X_Fraud['amount']-X_Fraud['oldBalanceDest']
X_Leg['errorBalanceDest']=X_Leg['newBalanceDest']+X_Leg['amount']-X_Leg['oldBalanceDest']
```

```
X['errorBalanceOrig']=X['newBalanceOrig']+X['amount']-X['oldBalanceOrig']
X['errorBalanceDest']=X['newBalanceDest']+X['amount']-X['oldBalanceDest']
```

Time Variations

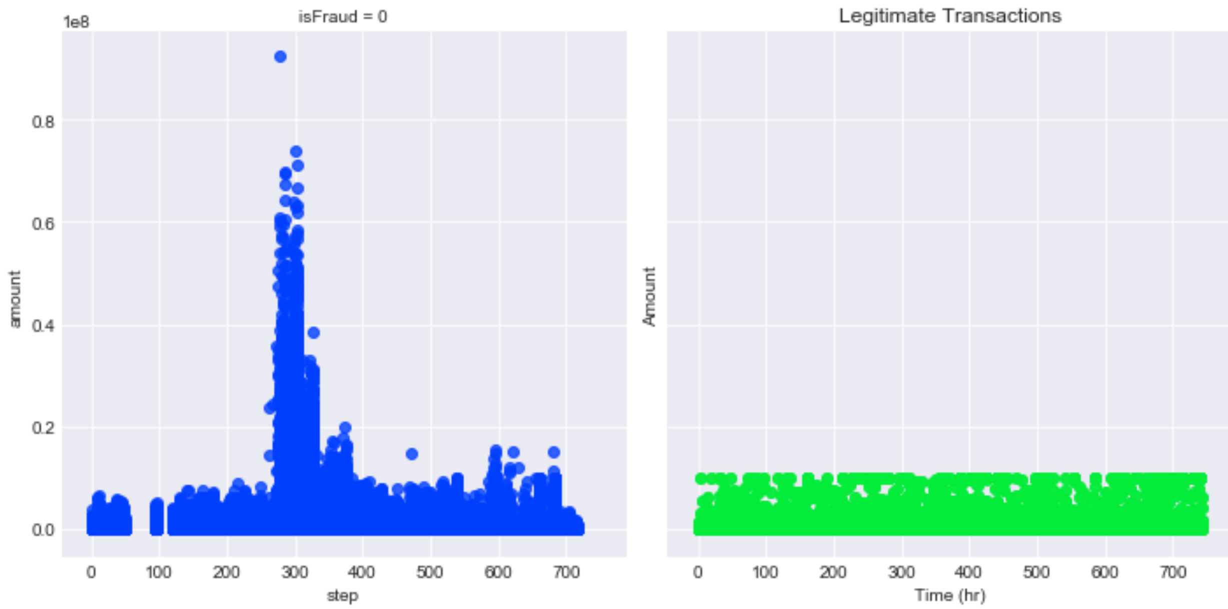
The changes in number of fraudulent and legitimate transactions over time are plotted next:





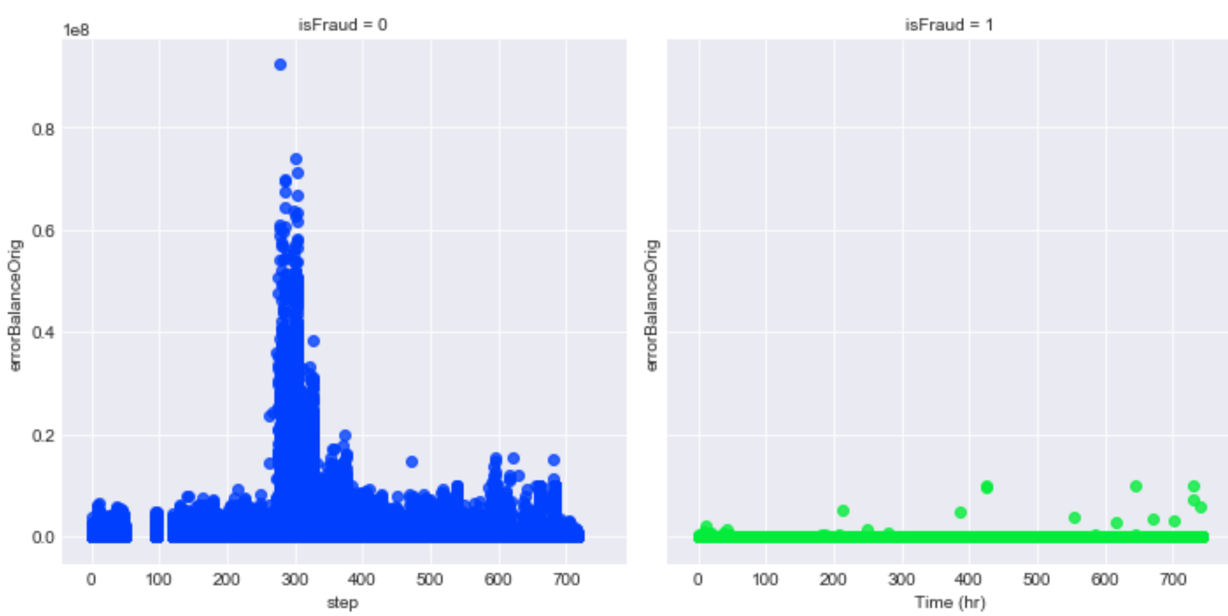
From the above figures, it is clear that for legitimate transactions, number of CASH-OUT is higher than TRANSFERs per hour, while for fraudulent transactions, CASH-OUT and TRANSFER are equal on many hours. Also, fraudulent cases are more distributed over time than legitimate transactions. For legitimate cases, both transactions drop after hour 400 (17 days).

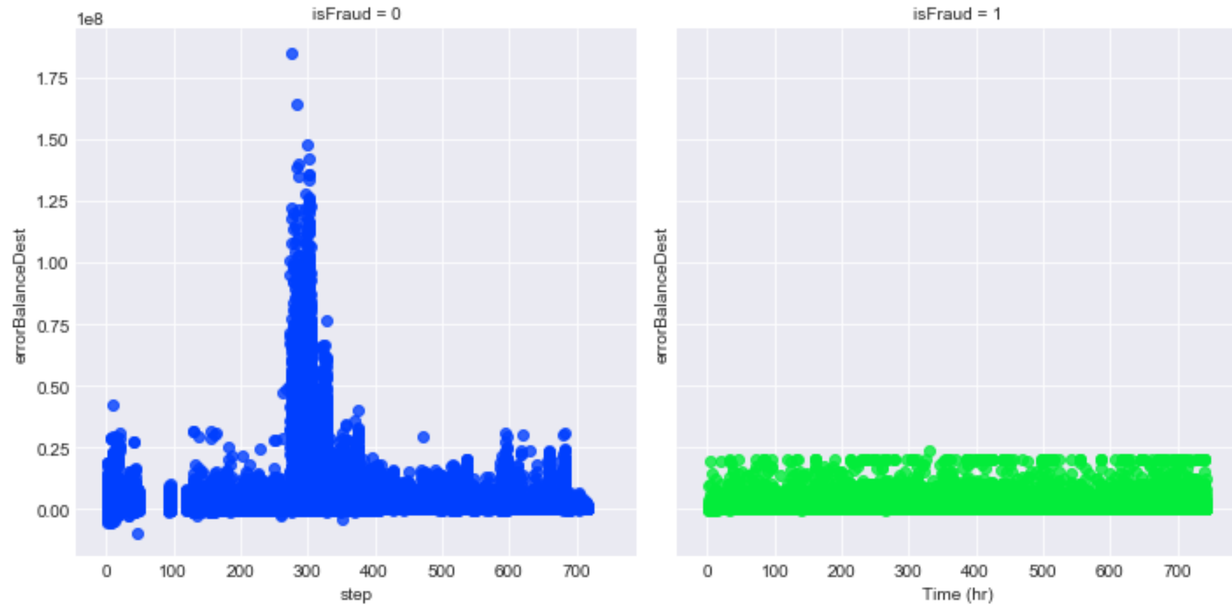
Amount variations over time is plotted below:

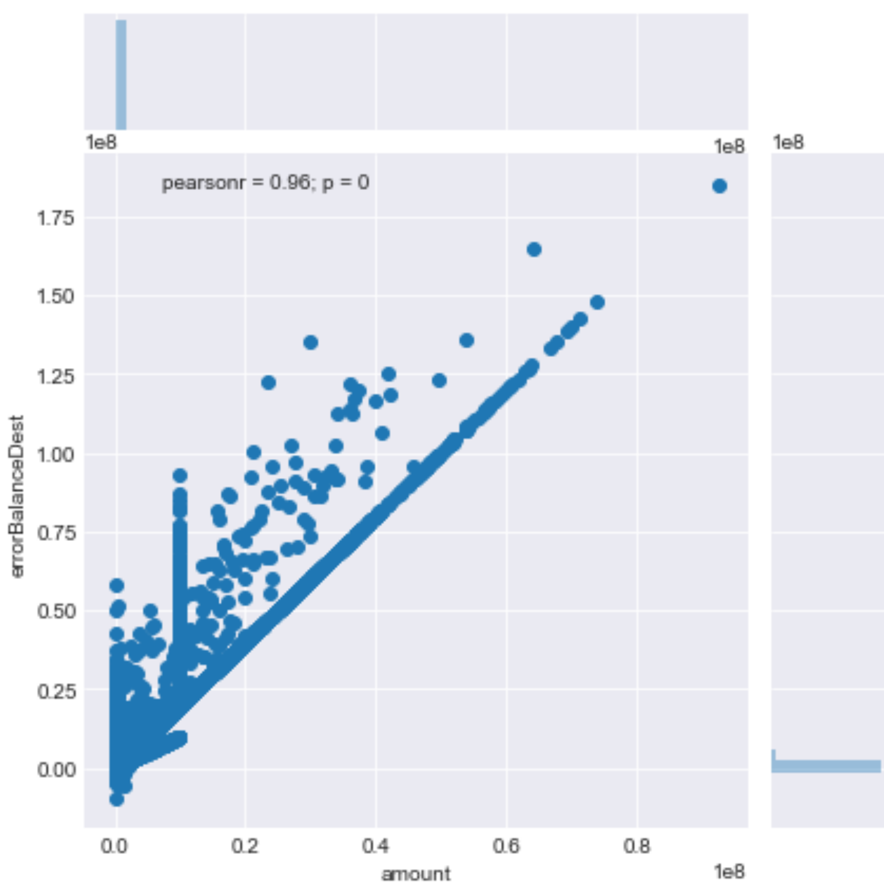


For legitimate transactions, there is a peak in amount at around 300hr, while for fraudulent cases, the amount is more evenly distributed.

The time variations of errorBalances are shown below. While the errorBalanceDest plots are very similar to amount, there is more clear distinction between fraudulent and legitimate cases for errorBalanceOrig values over time. This can also be seen on the correlation plot below.




isFraud = 0
isFraud = 1
1e8
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
errorBalanceDest
0
100
200
300
400
500
600
700
step
Time (hr)


pearsonr = 0.96; p = 0
1e8
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
errorBalanceDest
0.0
0.2
0.4
0.6
0.8
amount

The respective correlation of different features is shown on the heat map below. There is a high correlation between oldBalanceOrig and amount in the fraudulent transactions, which cannot be observed in the legitimate cases.

Inferential Statistics

In this section, some inferential statistics techniques are applied for the fraud detection data, in order to check whether or not the fraudulent and legitimate transactions are statistically different in the amount of transactions. The Null hypothesis is that there is no difference in amount of transactions between fraudulent and legitimate cases. Alternative hypothesis is that this amount is statistically different in fraudulent and legitimate transactions.

In order to perform this statistical evaluation, the mean of amount for fraudulent and legitimate cases are calculated, which are 1467967.30 and 314115.49, respectively. After calculating the standard deviation and size for each group, the standard error and z-score are obtained as follows:
standard error = 26534.76
z-score = 43.484

The p-value based on the above z-score is calculated as zero. Since the p-value is zero, we reject the null hypothesis. So there is a statistically significant difference between mean of amount for legitimate and fraudulent transactions.

Other Potential Datasets

Kaggle's credit card fraud detection data:

The datasets contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. It contains only numerical input variables which are the result of a PCA transformation. Unfortunately, due to confidentiality issues, the original features and more background information about the data is not provided.

German credit fraud dataset:

This dataset classifies people described by a set of attributes as good or bad credit risks. It contains categorical and integer data with 20 features and 1000 instances.

Initial Findings

The main technical challenge in predicting fraud is the highly imbalanced distribution between legitimate and fraudulent classes in 6 million rows of data. Another deficiency of this data stems from the possible discrepancies in its description and some redundant column values. The goal of this project is to solve these issues by a detailed data exploration and wrangling followed by choosing a suitable machine-learning algorithm to deal with the skew. Supervised classification algorithms will be used to predict fraudulent transactions.