Choose Your Own Project Submission

Johan Wikström

02 juni 2019

Contents

1 Introduction

For my own project I have choosen to dive into the world of fraud detection. Fraud detection is a very important business domain in which macine learning can be very helpful. In this domain, the data size is large, speeds a high, and there is dependancies between transactions.Despite the amount of data, creating good models that identify fraudulent behaviour is difficult, this because the proportion of fraudulent transaction usually is very low and very similar to non-fraudulent transaction.

I have choosen the paysim dataset from the Kaggle website. The dataset can be found here: https://www.kaggle.com/ntnu-testimon/paysim1. The dataset is a synthetic financial dataset for fraud detection. It is created from aggregated from an african mobile money service and is created to resemble normal transaction operations.

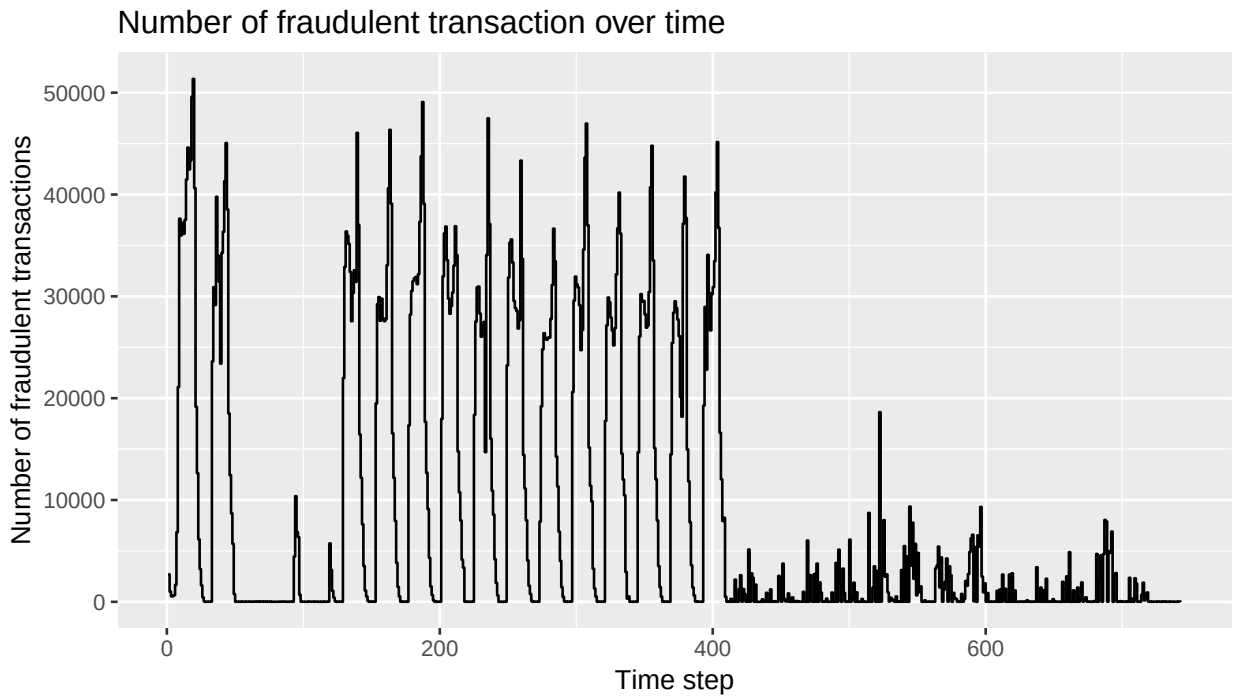
The dataset is a timeseries of transactions spanning over one month. The number of transaction is ~6.4 million and the proportion of fraud is only 0.12%. The dataset i made up of eleven attributs:

- step - maps a unit of time in the real world. In this case 1 step is 1 hour of time.
- 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 - This is the transactions made by the fraudulent agents inside the simulation.
- isFlaggedFraud - The business model aims to control massive transfers from one account to another and flags illegal attempts.

In this project, an analysis of the dataset and the fraudulent transactions is made. An important finding is that a majority of the fraudulent transaction is made in two steps: 1, Move the balance from one account to another; 2, emediately cash out the entire transaction. From this we are able to find fraudulent cash outs by looking and transfers made in the same hour and flag these two transactionos as suspected fraudulent transaction. After, a gradient boosting model was trained, with optimal paremeters estimated using k-fold cross validation, and since we have alot of date, downsampling was applied to handle the inbalance in the dataset. To handle the imbalance further, the metric we want to minimize is the F1 score.

2 Analysis and Methods

The first thing I did was to remove the isFlaggedFraud attribute since the attribute I want to predict is the IsFraud flag. Then, I looked at which steps we have fraudulent behaviour.



As we can see, there are alot of spikes of fraudulent transaction for the earlier steps, and later, the fradulent behaviour is more stable. Using this information, I split the data into training and validation set at a specific time step. This step was choosen to make the split as close to 80%-20% as possible. This splitting the data in this way was important since we are using a time series so using transaction from the future in training is not recommended.

The next step taken was to idenitfy which transaction types which are fraudulent. In the table below we can clearly see that all fraudulent transaction only comes from two types, TRANSFER and CASH_OUT. The proportion between them is very similar, which we will see the reason for later. This finding helps us to fitler away types which cannot be flagged as fraudulent in this specific case.

type	n_fraud
TRANSFER	4097
CASH_OUT	4116

Looking at the origin and destination of fraudulent transaction shows us that most destinations and origins appear only once with fraudulent transaction, only 44 destination are used twice for fraudulent transaction. This leads us to suspect the account information is not relevant. Since we are using time series data, a time series partiotioning would be prefereable when training. However, we are not using any account information, meaning that each transaction can be seen as individual. Thus, we can use a regular k-fold partitioning.

		Frequency of fraudulent Transactions	
		Origin	Destination
Number of transactions	1	8213	8125
	2	0	44

Lastly, I took a quick look at the fraudulent transaction. Doing so I found that the fraudulent transaction happen in a certain way. First the balance is transfered and then directly being extracted, which can be seen in the table below.

step	type	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDest	isFraud
1	TRANSFER	181.00	181.00	0.00	0.00	0.00	TRUE
1	CASH_OUT	181.00	181.00	0.00	21182.00	0.00	TRUE
1	TRANSFER	2806.00	2806.00	0.00	0.00	0.00	TRUE
1	CASH_OUT	2806.00	2806.00	0.00	26202.00	0.00	TRUE
1	TRANSFER	20128.00	20128.00	0.00	0.00	0.00	TRUE
1	CASH_OUT	20128.00	20128.00	0.00	6268.00	12145.85	TRUE
1	CASH_OUT	416001.33	0.00	0.00	102.00	9291619.62	TRUE
1	TRANSFER	1277212.77	1277212.77	0.00	0.00	0.00	TRUE
1	CASH_OUT	1277212.77	1277212.77	0.00	0.00	2444985.19	TRUE
1	TRANSFER	35063.63	35063.63	0.00	0.00	0.00	TRUE
1	CASH_OUT	35063.63	35063.63	0.00	31140.00	7550.03	TRUE
1	TRANSFER	25071.46	25071.46	0.00	0.00	0.00	TRUE
1	CASH_OUT	25071.46	25071.46	0.00	9083.76	34155.22	TRUE
1	CASH_OUT	132842.64	4499.08	0.00	0.00	132842.64	TRUE
1	TRANSFER	235238.66	235238.66	0.00	0.00	0.00	TRUE
1	CASH_OUT	235238.66	235238.66	0.00	0.00	235238.66	TRUE
2	TRANSFER	1096187.24	1096187.24	0.00	0.00	0.00	TRUE
2	CASH_OUT	1096187.24	1096187.24	0.00	0.00	1096187.24	TRUE
2	TRANSFER	963532.14	963532.14	0.00	0.00	0.00	TRUE
2	CASH_OUT	963532.14	963532.14	0.00	132382.57	1095914.71	TRUE

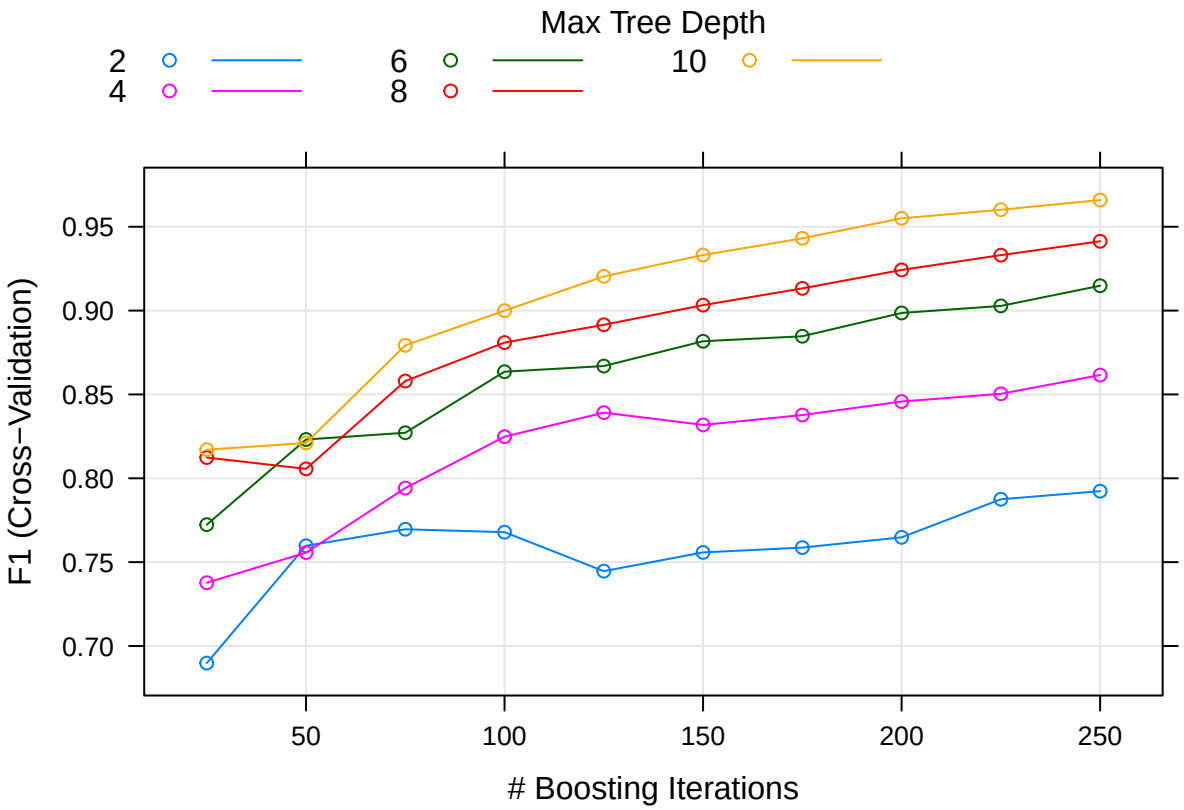
This lead me to creating an algorithm for calculating a flag, marking a transaction as possibly fraudulent. The algorithm for this was: When a CASH_OUT is being made, check the last hour for transactions of the same amount (Accounts seem to be missmatched or erronous, thus cannot be used to trace the balance). If there is a match, set this flag to TRUE. Then, proceed to mark the TRANSFER made that lead to the CASH_OUT as possibly fraudulent. This approach is intuative to me since the TRANSACTION it self does not seem to cause concern until it leads to a fraudulent transaction. In some sense, it is a sort of back tracing, marking transaction leading to a possibly fraudulent transaction as suspected fraudulent.

Since we want to identify as many fraudulent transactions as possible without predicting false positives, the F1 score was used as the evaluation metric. Also, 5-fold cross validation was used together with a parameter grid to find the optimal set of parameters for the Gradient Boosting Model. The grid used was: $interaction.depth \in [2, 4, .., 10$, $n.trees \in [25, 50, ..250]$, $Shrinkage = 0.1$, $n.minobsinnode = 10$.

3 Results

The data was split at step 354, resulting in a split of 19,94 % of the dataset being the validation set.

The best model was reached at 250 boosting iterations, reaching a F1 score 0,97. The plot below shows the F1 score depending on boosting iteration and max tree depth.



The best model was found using these parameters:

shrinkage	interaction.depth	n.minobsinnode	n.trees	F1	F1SD
0.10	10.00	10.00	250.00	0.97	0.01

Further, we can see in the table below that the suspected fraudulent flag, isSusp, is very important for the model.

	Overall
step	0.12
type	0.05
amount	0.53
oldbalanceOrg	0.73
newbalanceOrig	0.00
oldbalanceDest	0.23
newbalanceDest	0.06
isSusp	100.00

Applying the model on the validation data, we get a F1 score 0,997.By looking at the confussion matrix, we can see that we hae only 24 wrongfully predictions.

	FALSE	TRUE
FALSE	548236	14
TRUE	10	4244

4 Conclusion and Discussion

In conclussion, we have seen the difficulty in fraud detection. The data is big, dependant, and often fraudulent transactions are not easily identified when compared to non-frauduelent ones. To be able to identify fraudulent transaction, knowledge about how to create features like flags is crucial for success.

In this project, I found an algorithm of flagging suspected fraudulent transactions when funds are moved between accounts and cashed out at the same time. By marking the transaction and cash out as suspected fraudulent transactions led to a great success in identifying fraudulent fransaction.

Further work could be done in creating more features and possibly choose other sampling methods to handle the large imbalance. In the bigger picture, this algorithm can only be applied to this specific case. Therefore, in a business system, more models should be created and/or integrated to identify other fraudulent behaviours.