A Project Report

On

Fraud Risk Management

Submitted by

Austin Paul

Supervised by

Mr. Sujith Nair
Mr. Manish Mer

Date: 27/06/2023

Place: Concerto Software & Systems, Mahape

Abstract

MLE: Maximum Likelihood Estimation

It is important to recognize fraudulent credit card transactions so that customers are not charged for items that they did not purchase and to prevent the fraudulent customers from making transactions.

Many models were created to detect the potential risks in the transaction where the results were binary, that is 0(valid) and 1(fraud).

The MLE approach applies a modified binary multivariate logistic analysis to model dependent variables to determine the expected probability of success of belonging to a certain group.

For instance, given a set of independent variables (e.g., Card holder, Merchant ID, Terminal ID, Cust IP), we can model the probability of default using MLE.

Probability of risk measures the degree of likelihood that the transaction will result as charge backs. The higher the risk probability, the higher the chances that the following transactions would result as a failed transaction result or may result in charge back.

This document details my approach to the problem of calculating the risk of the transactions. I have developed an API which can be directly which involves calculating the risk with all the preprocessing required.

Proposed System

Problem Statement

Making a model to predict the probability of risks in real time transactions.

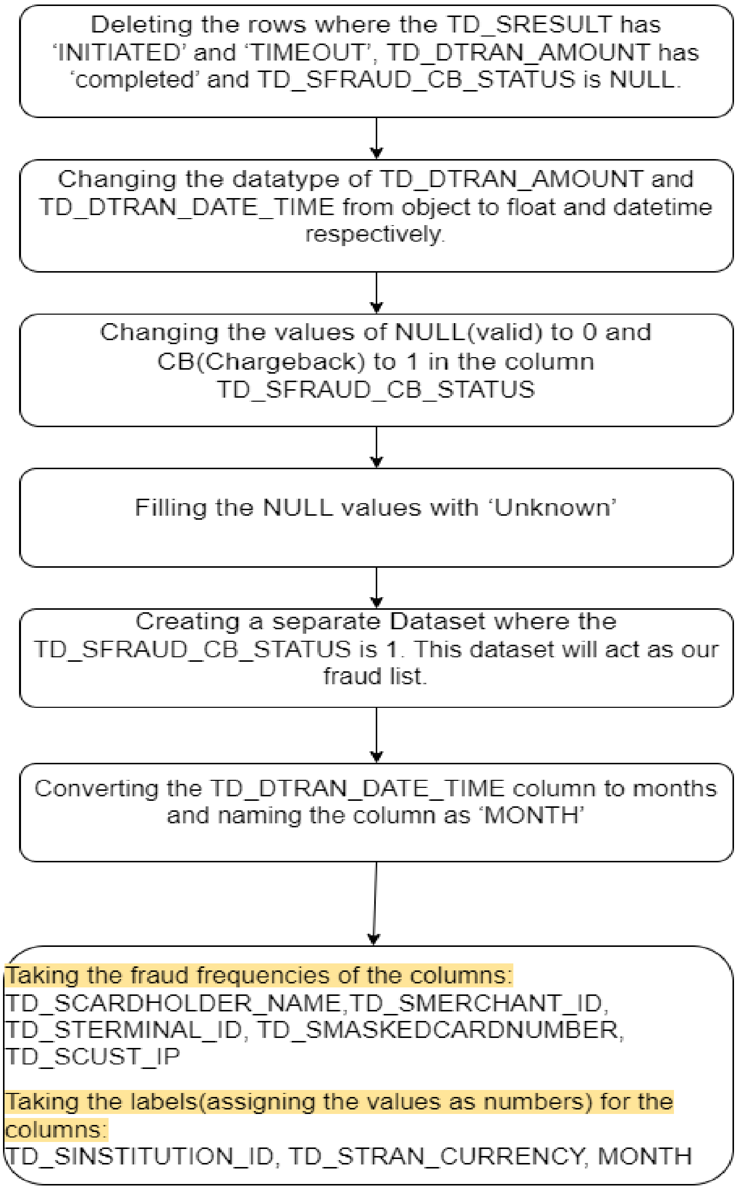
Design of the System

The given dataset contains more than 46,000 records with 93 columns.

The selected columns that were pre-processed were:

TD_SCARDHOLDER_NAME, TD_SMERCHANT_ID, TD_STERMINAL_ID, TD_SINSTITUTION_ID, TD_SMASKEDCARDNUMBER, TD_SCUST_IP, TD_DTRAN_DATE_TIME, TD_SRESULT, TD_STRAN_CURRENCY, TD_DTRAN_AMOUNT and TD_SFRAUD_CB_STATUS

Data preprocessing includes:



Deleting the rows where the TD_SRESULT has 'INITIATED' and 'TIMEOUT', TD_DTRAN_AMOUNT has 'completed' and TD_SFRAUD_CB_STATUS is NULL.

Changing the datatype of TD_DTRAN_AMOUNT and TD_DTRAN_DATE_TIME from object to float and datetime respectively.

Changing the values of NULL(valid) to 0 and CB(Chargeback) to 1 in the column TD_SFRAUD_CB_STATUS

Filling the NULL values with 'Unknown'

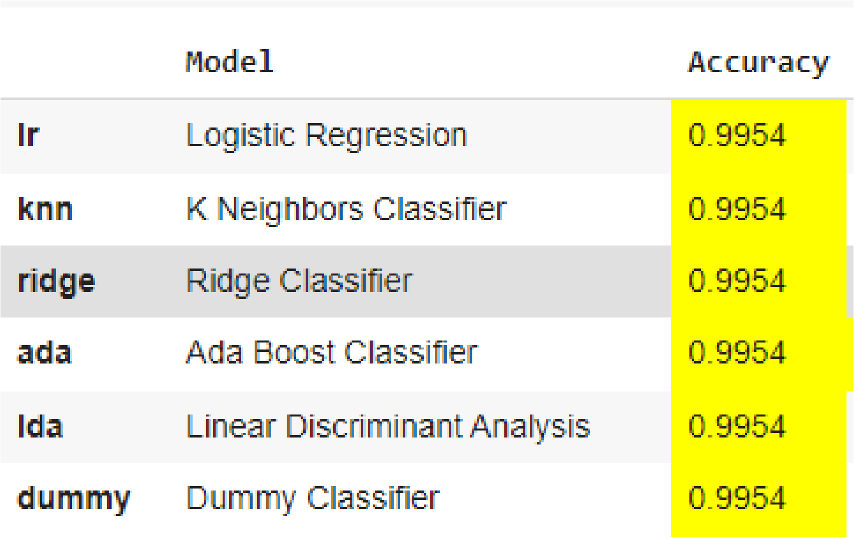
Creating a separate Dataset where the TD_SFRAUD_CB_STATUS is 1. This dataset will act as our fraud list.

Converting the TD_DTRAN_DATE_TIME column to months and naming the column as 'MONTH'

Taking the fraud frequencies of the columns:
TD_SCARDHOLDER_NAME,TD_SMERCHANT_ID,
TD_STERMINAL_ID, TD_SMASKEDCARDNUMBER,
TD_SCUST_IP

Taking the labels(assigning the values as numbers) for the columns:
TD_SINSTITUTION_ID, TD_STRAN_CURRENCY, MONTH

MODELS FOR FRAUD DETECTION:



	Model	Accuracy
lr	Logistic Regression	0.9954
knn	K Neighbors Classifier	0.9954
ridge	Ridge Classifier	0.9954
ada	Ada Boost Classifier	0.9954
lda	Linear Discriminant Analysis	0.9954
dummy	Dummy Classifier	0.9954

MODELS FOR CALCULATING PROBABILITY:

1. SIMPLY TAKING PERCENTAGE

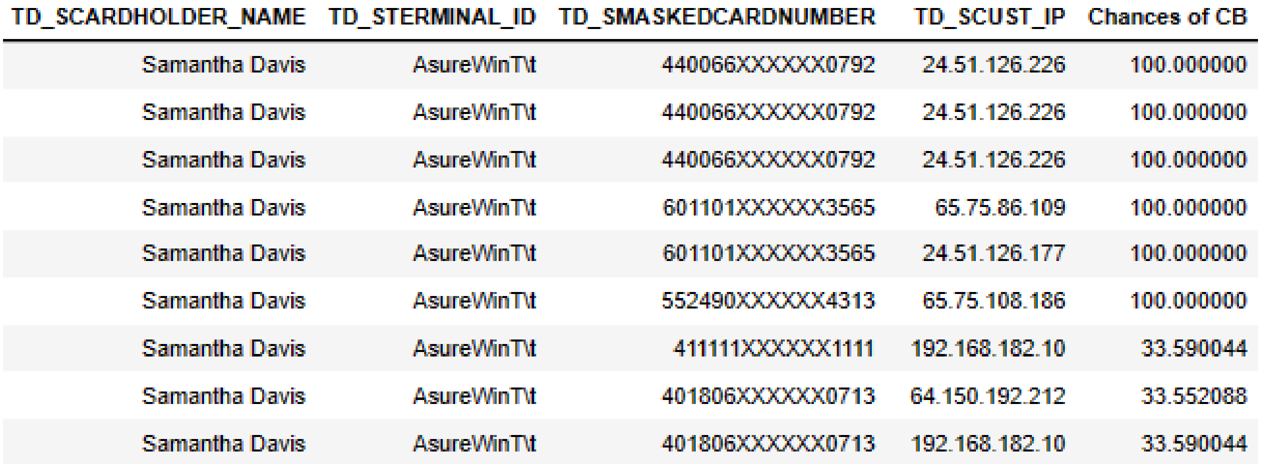
PERCENTAGE = SUM (frequency of the value in the fraud list / frequency of the value in actual data) * 100

Added other parameters for analysis:

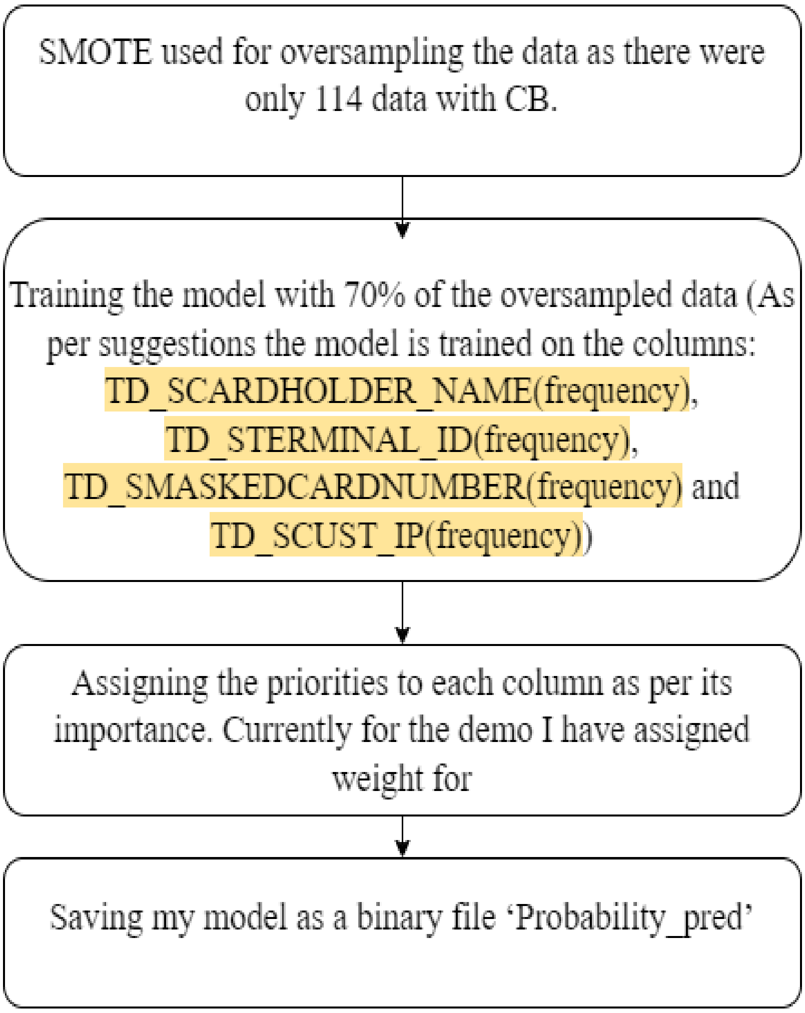
1. Card numbers with the same card holder name in the fraud list and in actual data.
2. Past transaction results of the card holder.
3. Card numbers with the merchant in the fraud list.
4. Card holders with the merchant in the fraud list.
5. Card numbers with the same IP name in the fraud list and in actual data.
6. Card holders with the same IP in the fraud list.

2. MODEL FOR CALCULATING PROBABILITY (IN USE)

Made a model for calculating probability using Most Likelihood Estimation.



TD_SCARDHOLDER_NAME	TD_STERMINAL_ID	TD_SMASKEDCARDNUMBER	TD_SCUST_IP	Chances of CB
Samantha Davis	AsureWinT\t	440066XXXXXX0792	24.51.126.226	100.000000
Samantha Davis	AsureWinT\t	440066XXXXXX0792	24.51.126.226	100.000000
Samantha Davis	AsureWinT\t	440066XXXXXX0792	24.51.126.226	100.000000
Samantha Davis	AsureWinT\t	601101XXXXXX3565	65.75.86.109	100.000000
Samantha Davis	AsureWinT\t	601101XXXXXX3565	24.51.126.177	100.000000
Samantha Davis	AsureWinT\t	552490XXXXXX4313	65.75.108.186	100.000000
Samantha Davis	AsureWinT\t	411111XXXXXX1111	192.168.182.10	33.590044
Samantha Davis	AsureWinT\t	401806XXXXXX0713	64.150.192.212	33.552088
Samantha Davis	AsureWinT\t	401806XXXXXX0713	192.168.182.10	33.590044



SMOTE used for oversampling the data as there were only 114 data with CB.

Training the model with 70% of the oversampled data (As per suggestions the model is trained on the columns: TD_SCARDHOLDER_NAME(frequency), TD_STERMINAL_ID(frequency), TD_SMASKEDCARDNUMBER(frequency) and TD_SCUST_IP(frequency))

Assigning the priorities to each column as per its importance. Currently for the demo I have assigned weight for

Saving my model as a binary file ‘Probability_pred’

After calculating the chances of risk it stores the result in the database in JSON format.

The database was created using mongo db.

Added other parameters for analysis:

1. Card numbers with the same card holder name in the fraud list and in actual data.
2. Past transaction results of the card holder.
3. Card numbers with the merchant in the fraud list.
4. Card holders with the merchant in the fraud list.
5. Card numbers with the same IP name in the fraud list and in actual data.
6. Card holders with the same IP in the fraud list.

OUTCOME :

My model calculates risk based on the past historical data where the values in the column TD_SFRAUD_CB_STATUS were considered NULL as valid transactions and Chargeback as fraud transactions. I also created a separate fraud list where all the transactions had their status as Chargeback.

When a new transaction enters in real time, it undergoes preprocessing where the columns are replaced with frequencies or labels.
FREQUENCY = (count of values in fraud list / count of values in historical transactions)

The new DataFrame goes through their assigned weights. We can input the weights for the columns to decide the priority of that column, with this if a certain column had their frequency less than others but plays an important role in fraud transactions can have its magnitude higher with the assigned weights.

Then the model calculates the probability and returns to the user.

Currently the performance of the model is good.

FUTURE SCOPE:

We can integrate our model with an API which can be used in web services and directly calculate the probability.

We can have more columns for evaluation according to the requirement in the future.