

ITI105 Final Project Presentation

Financial Fraud Detection using Machine Learning

Adelene Ng, Gareth Jonathan Halim

Problem Statement

- Objective: To predict if a financial transaction is fraudulent or not
- Binary Classification problem
- Ideal outcome: To correctly predict a transaction which is fraudulent as actually fraudulent

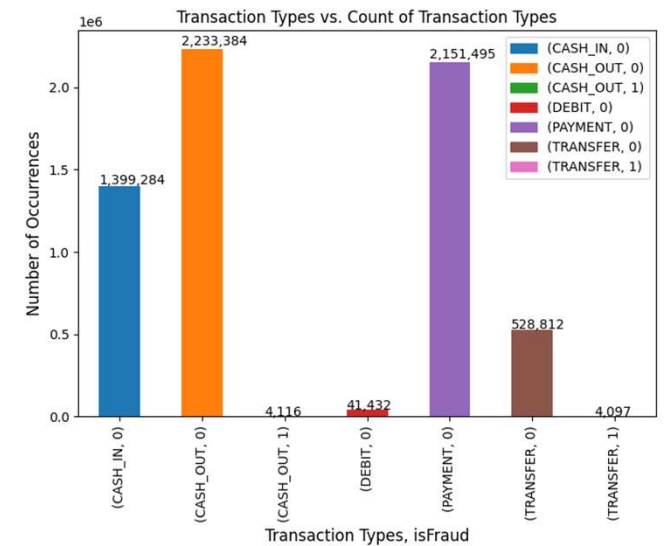
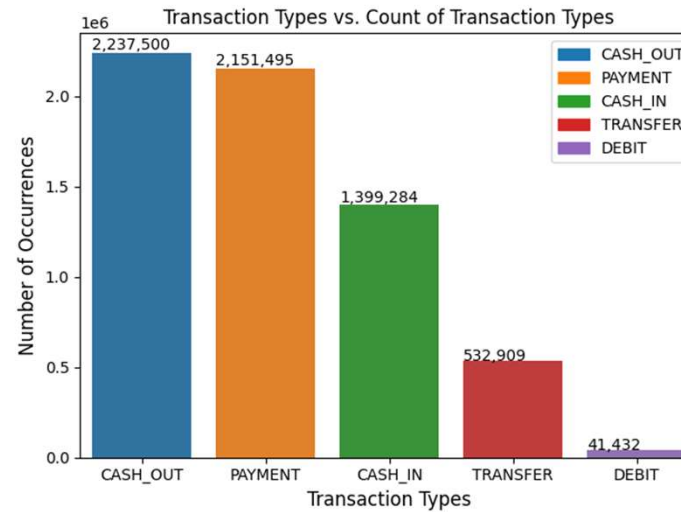
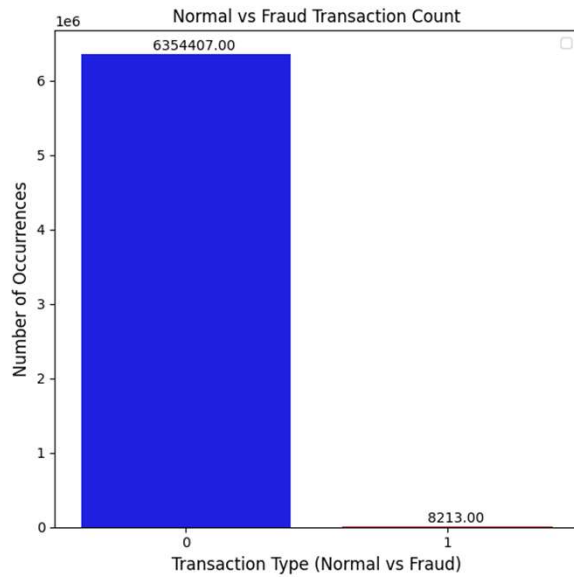
Datasets

- From Kaggle, generated by the PaySim mobile generator
- Based on a sample of real transactions extracted from one month of financial logs from a mobile money service

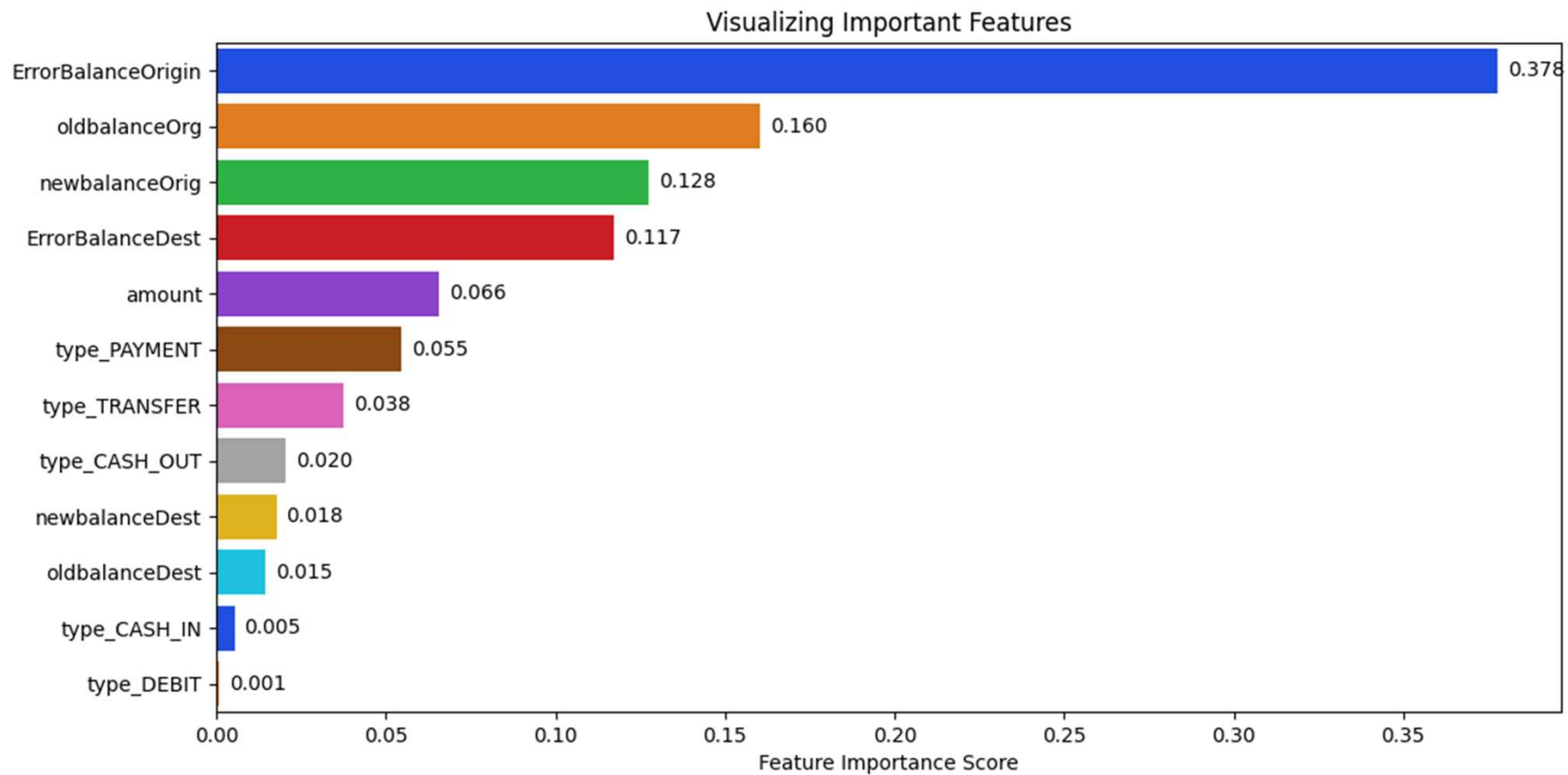
Dataset Features

Features	Description
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
oldbalanceOrig	Initial balance before the transaction
newbalanceOrig	New balance after the transaction
nameDest	Customer who is the recipient of the transaction
oldbalanceDest	Initial balance of recipient before the transaction
newbalanceDest	New balance of recipient after the transaction
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 of 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.

Exploratory Data Analysis



Feature Importance



Experiments

Algorithms	recall	precision	f1	balanced accuracy
lr	0.943	0.023	0.044	0.945
xgb	0.996	0.826	0.903	0.998
nb	0.993	0.003	0.006	0.783
dt	0.995	0.975	0.985	0.997
rfc	0.995	0.867	0.927	0.997
cb	0.996	0.984	0.99	0.998
lgbm	0.998	0.808	0.893	0.999
bg	0.995	0.858	0.921	0.997
Maximum	0.998	0.984	0.99	0.999

- LightGBM has a recall of **0.998** compared to CatBoost, whose recall is **0.996**.
- CatBoost has a higher precision **0.984** compared to LightGBM **0.808**.
- The F1-score for CatBoost is **0.99** whilst that of LightGBM is **0.893**.
- We can conclude that **CatBoost is the better classifier.**

Hyper-parameter Tuning: Important – Controls Overfitting

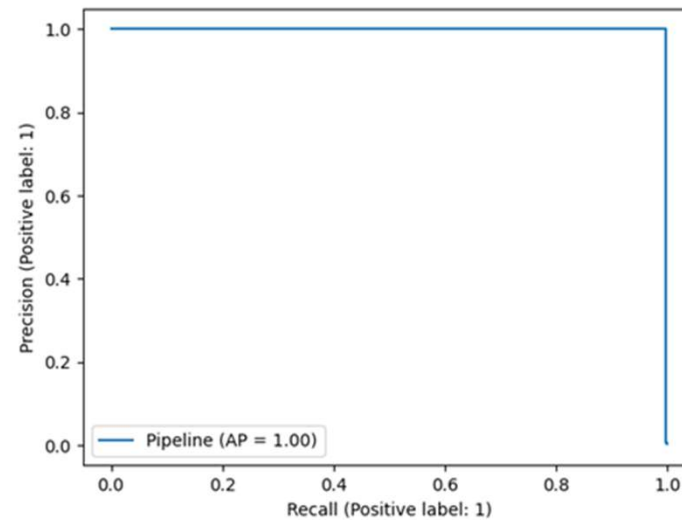
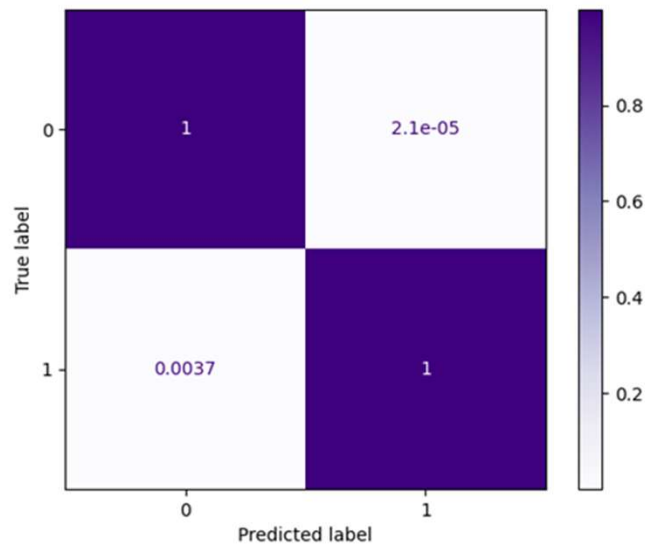
CatBoost	Values		LightGBM	Values	
Learning Rate (This determines how fast or slow the model will learn. The default is usually 0.03)	0.001, 0.01, 0.5	0.01	Learning Rate (Default: 0.1)	0.001, 0.01, 0.5	0.01
Depth (up to 16) (tree depth)	4, 6, 10	10	Max depth (Controls the max depth of each trained tree; If you use a large value of max_depth , your model will likely be over fit to the train set)	10, 50	10
L2-leaf-reg (Coefficient at the L2 regularization term of the cost function. The default is 3.0)	10, 15, 25	10	Min Data in Leaf: the minimum number of data/sample/count per leaf (default is 20; lower min_data_in_leaf means less conservative/control, potentially overfitting).	500, 1000	1000

Results before/after hyper-parameter tuning

Score Type	CatBoost (Old Hyper Parameters)	CatBoost	Light GBM (Old Hyper Parameters)	Light GBM
Precision Score	0.984	0.993	0.808	0.888
Recall Score	0.996	0.996	0.998	0.997
F1 Score	0.99	0.995	0.893	0.939
Balanced Accuracy	0.998	0.998	0.999	0.998
AUC ROC Score	0.998	0.998	0.999	0.998

- LightGBM has a recall of **0.997** compared to CatBoost, whose recall is **0.996**.
- CatBoost has a new higher precision **0.993** compared to LightGBM's new precision score of **0.888**.
- The new F1-score for CatBoost is **0.995** whilst the new F1-score for LightGBM is **0.939**.
- We conclude that **CatBoost is the better classifier**.

CatBoost (Before hyper-parameter tuning)

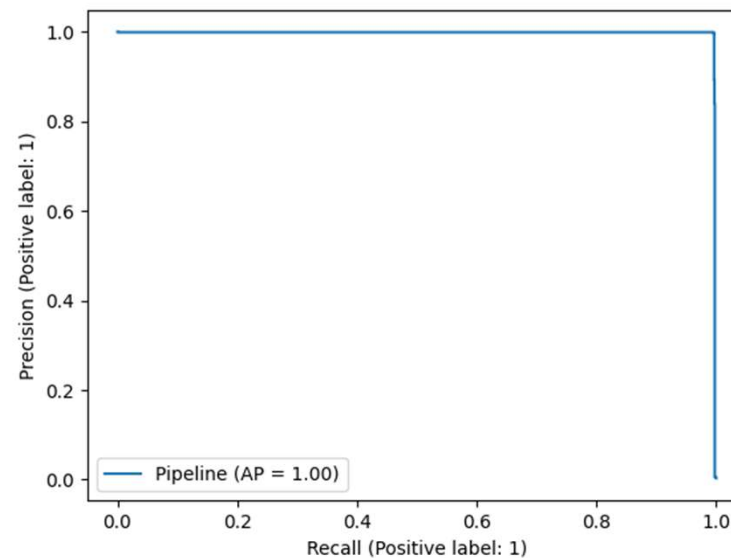
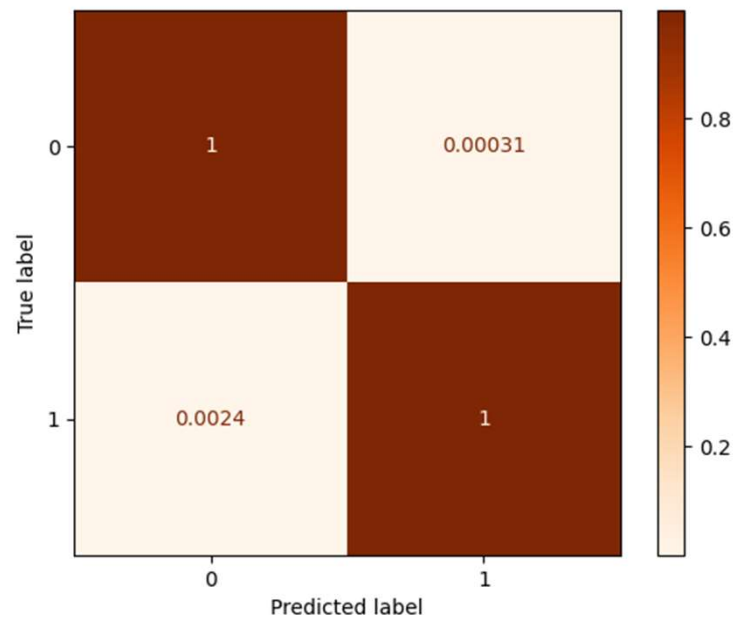


Hyper-parameter	Value
iterations	5
learning_rate	0.1
random_state	88



near perfect precision recall curves

LightGBM (Before hyper-parameter tuning)



Hyper-parameter	Value
boosting_type	gbdt
learning_rate	0.1
random_state	88



near perfect precision recall curves

CatBoost and LightGBM Classification Reports

CatBoost

	precision	recall	f1-score	support
--	-----------	--------	----------	---------

0	1.00	1.00	1.00	1270881
1	0.99	1.00	0.99	1643

accuracy			1.00	1272524
macro avg	1.00	1.00	1.00	1272524
weighted avg	1.00	1.00	1.00	1272524

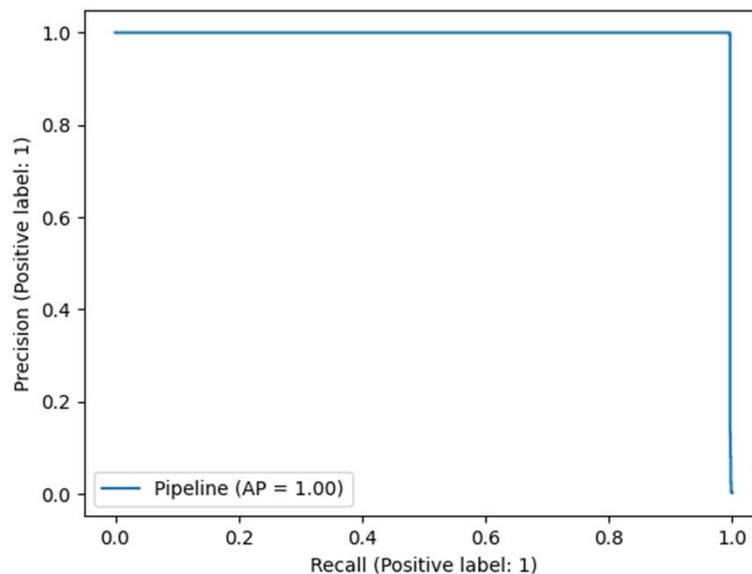
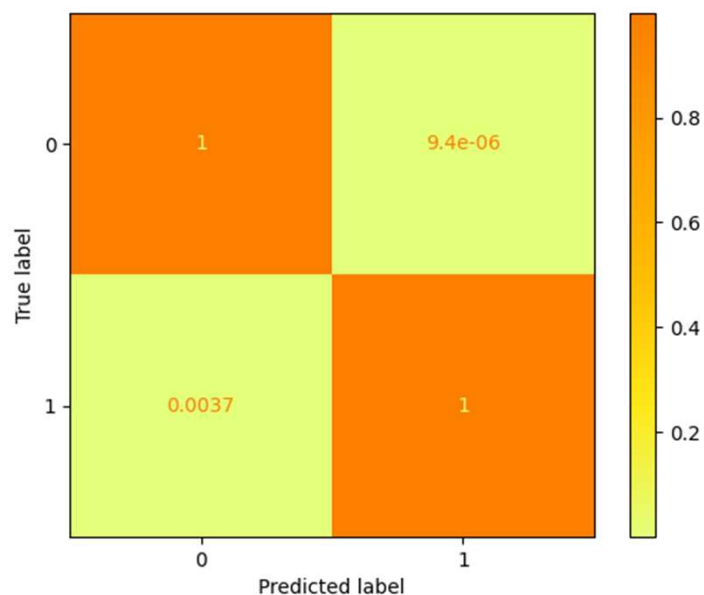
LightGBM

	precision	recall	f1-score	support
--	-----------	--------	----------	---------

0	1.00	1.00	1.00	1270881
1	0.89	1.00	0.94	1643

accuracy			1.00	1272524
macro avg	0.94	1.00	0.97	1272524
weighted avg	1.00	1.00	1.00	1272524

CatBoost (applying new hyper-parameters)

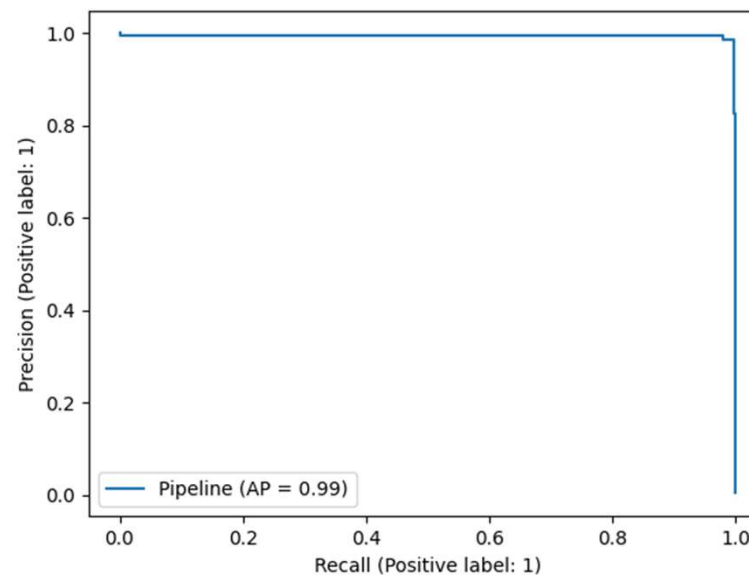
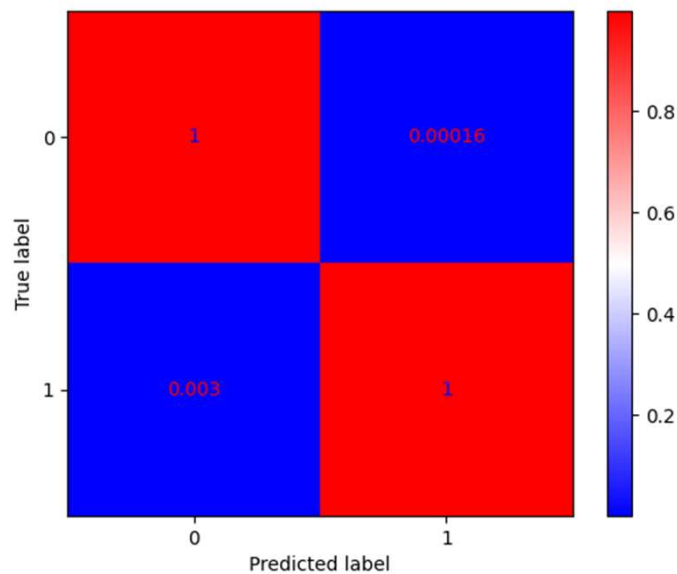


Hyper-parameters	Value
Learning Rate	0.01
Depth	10
L2-leaf-reg	10



near perfect precision recall curves

LightGBM (applying new hyper-parameters)



near perfect precision recall curves

Hyper-parameters	Value
Learning Rate	0.01
Max Depth	10
Min Data in leaf	1000

Web Application UI

Fraud Detection Service

Transaction Amounts			
Amount: <input type="text" value="181.0"/>			
Balances			
Old Balance Original: <input type="text" value="0.0"/>	Old Balance Destination: <input type="text" value="0.0"/>	New Balance Original: <input type="text" value="0.0"/>	New Balance Destination: <input type="text" value="0.0"/>
Transaction Type			
Cash In: <input type="radio"/> Cash Out: <input type="radio"/> Debit: <input type="radio"/> Payment: <input type="radio"/> Transfer: <input checked="" type="radio"/>			
<input type="button" value="Submit"/>			

Fraud Detection Results

Verdict: Transaction NORMAL

Fraud Detection Results

Verdict: Transaction is FRAUDULENT

