# CASE STUDY

ABDULLAHI A. IBRAHIM

Task

- Predict income and expenses for a holdout sample of ~10k users for the month of August based on a training sample of ~10ks from February – July.

- Based on your judgement of the usefulness of the results, either aggregate the data into incoming & outgoing flows, or predict base on the transaction type/category level
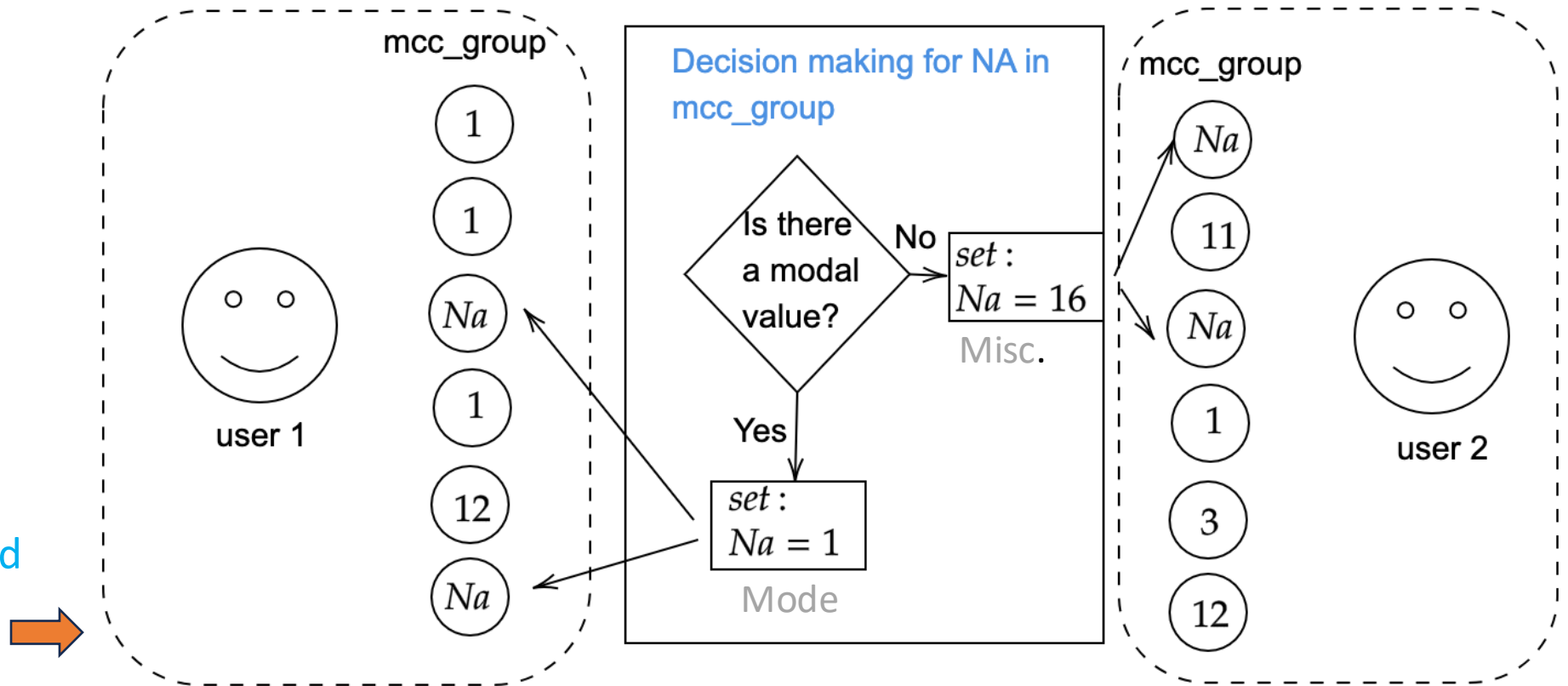
Assumptions

1. Aggregate data into In (incoming) and Out (outgoing) flows.

Direction

BBU
CT
DR

In
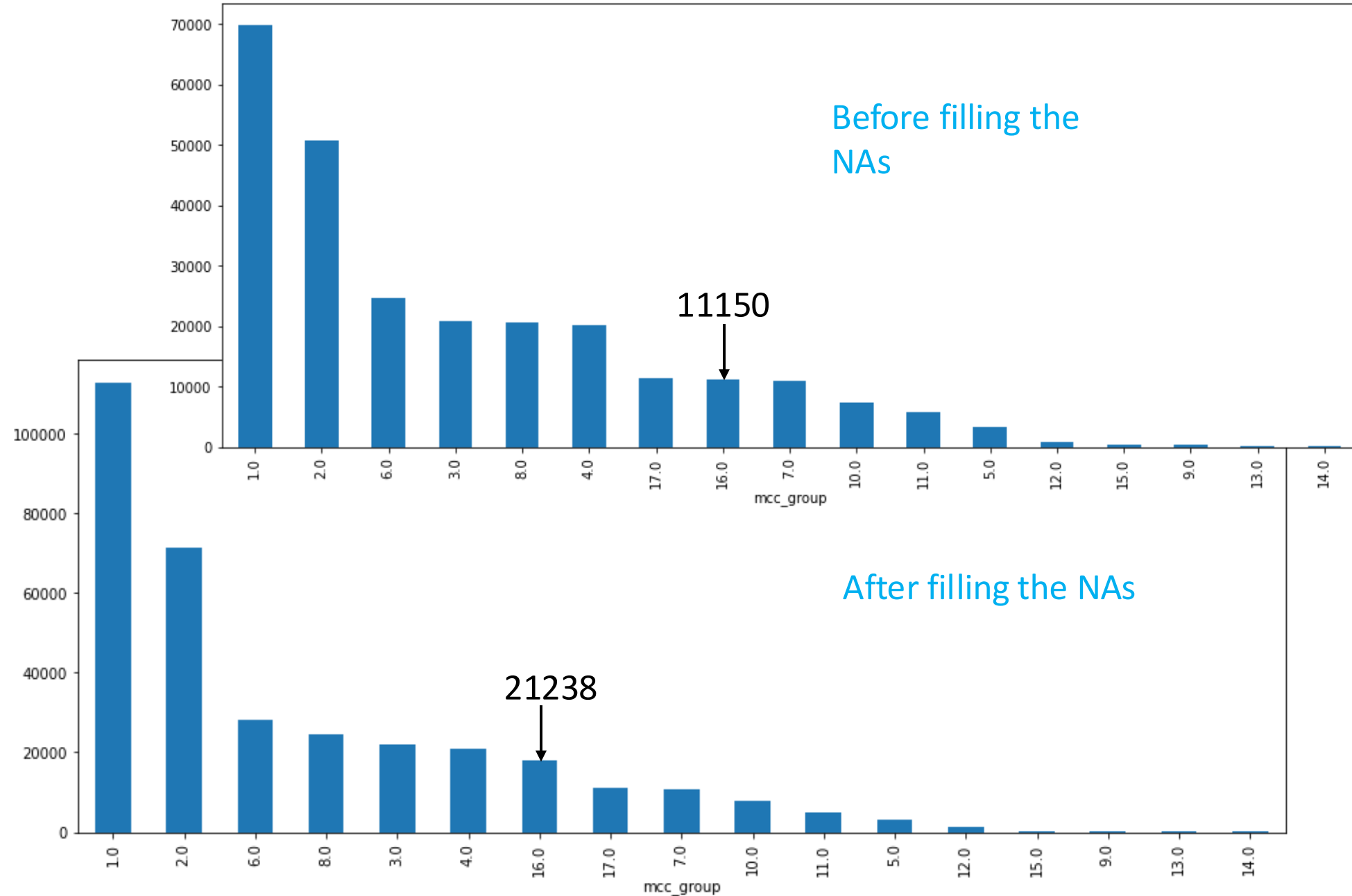
0

1

Out

BUB
DD
DT
FT
PT
TUB
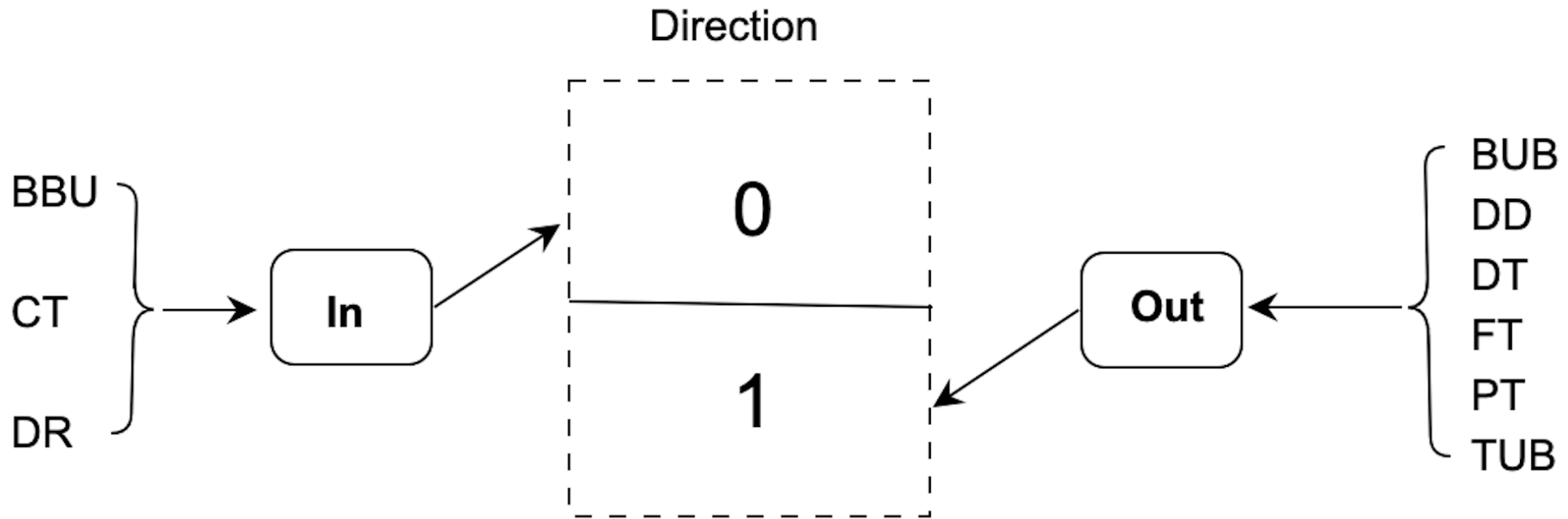
Fill missing values in mcc_group
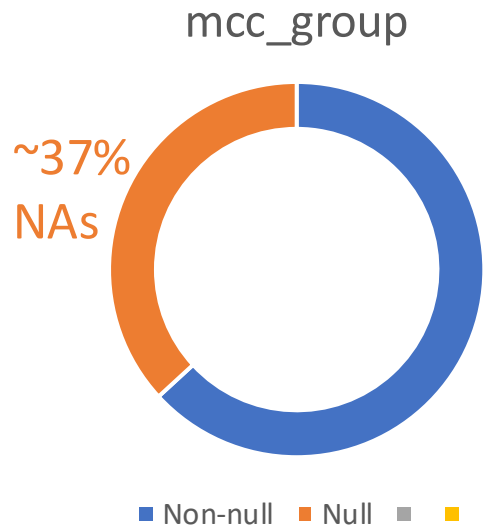
Impact of filling NAs

After filling the Nas
The frequency of
each class increased
without favouring
any group

Before filling the
NAs

After filling the NAs
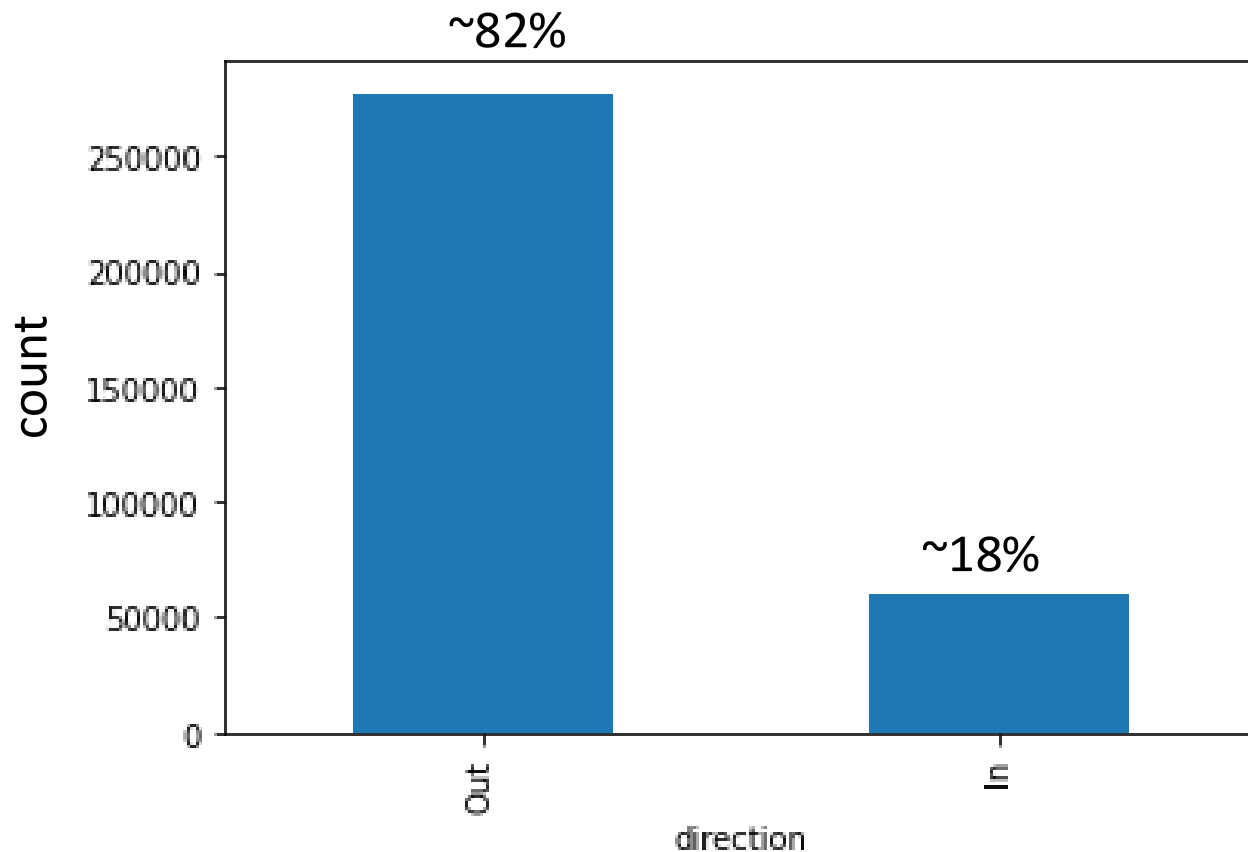
Feature engineering

New Features
- Transaction Agent
- Transaction direction
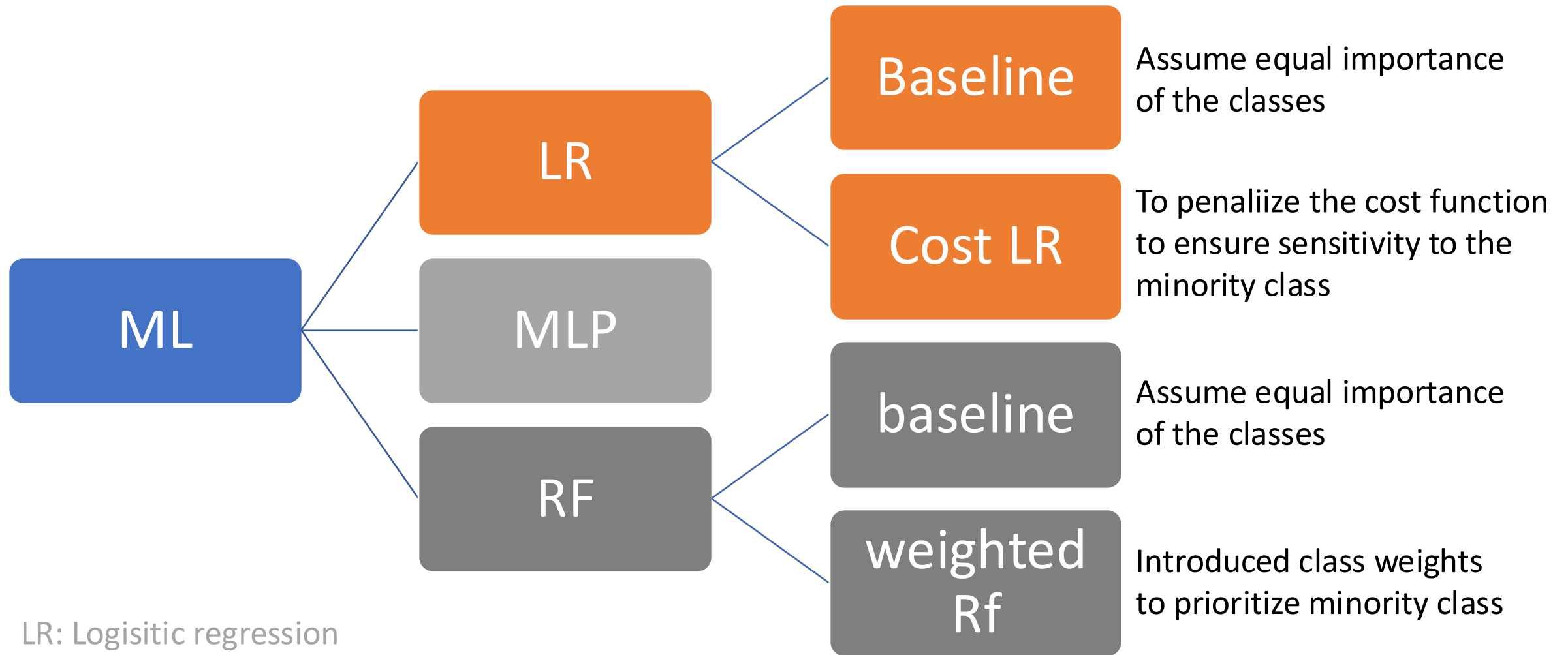- Transaction month and day

Feature preprocessing
- Encoding categorical features
- Normalization
- Train/valid/test split

Data distribution & split

- Shows imbalance distribution

- To avoid skewed result towards Out group, we do **not split randomly**

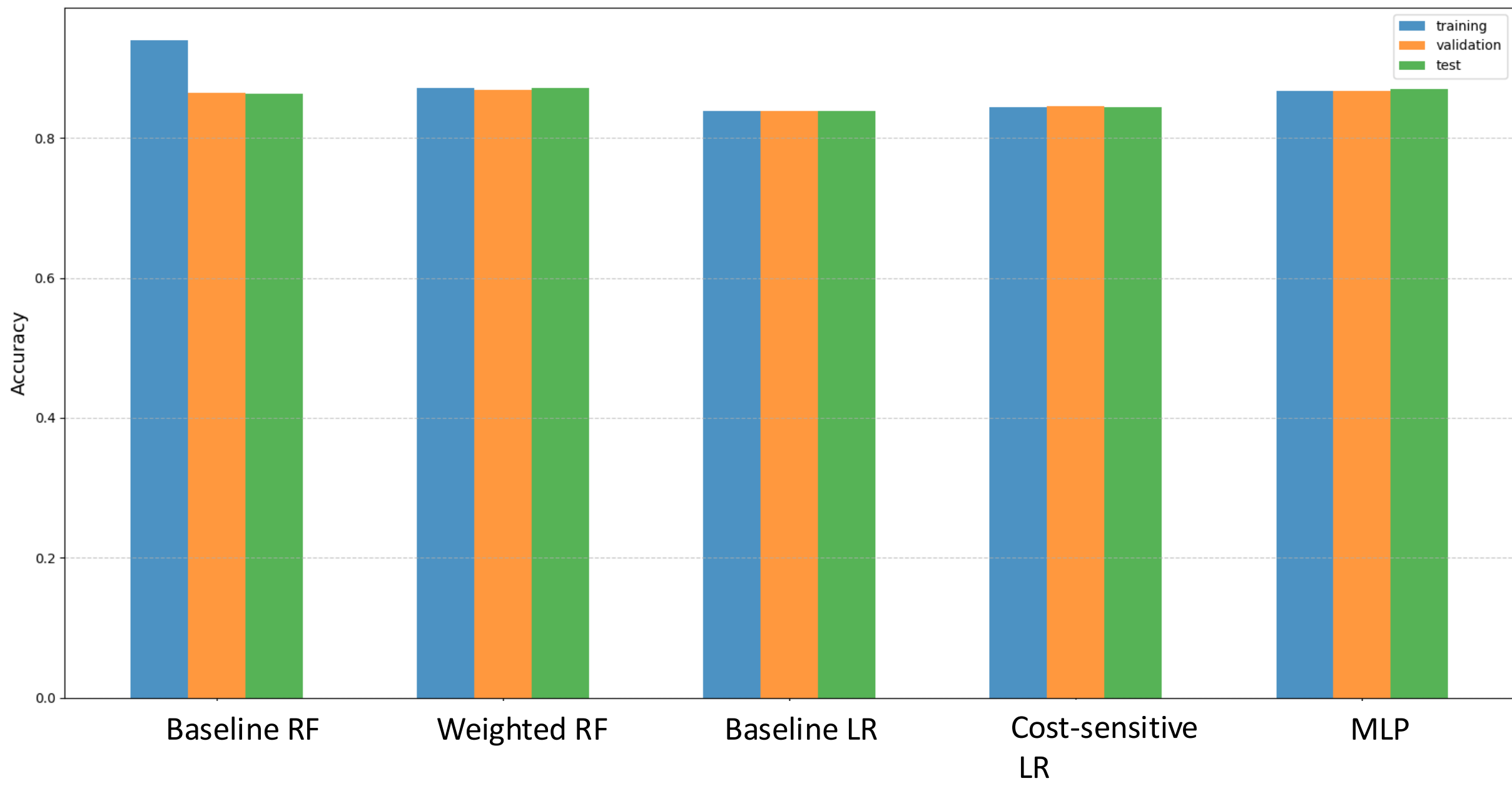- Instead, we maintained this proportion across training, validation and test sets

ML methods

ML

LR

MLP

RF

Baseline — Assume equal importance of the classes

Cost LR — To penaliize the cost function to ensure sensitivity to the minority class

baseline — Assume equal importance of the classes

weighted Rf — Introduced class weights to prioritize minority class
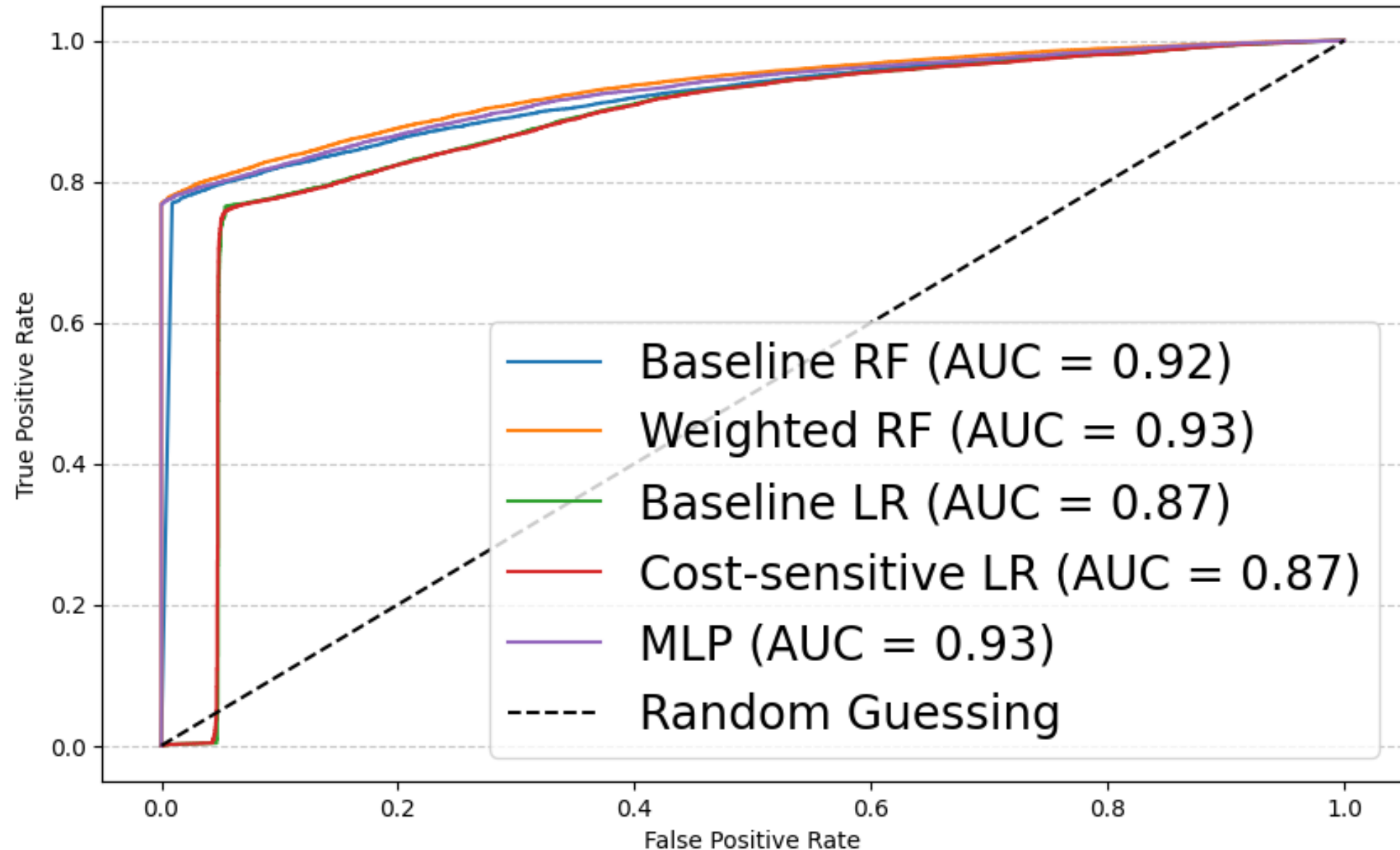
LR: Logisitic regression
RF: Random forest
MLP: Multilayer perceptron

Accuracy of the models

Sensitivity of the models

Summary

- Our goal is to predict income and expenses of a user transaction.

- To do this we compare the performace of different model (baseline and extended versions)

- Our result shows that by adding penalities/weight the model performed better than the baseline model.

- The best performing model(s) achieved upto 87% accuracy, 93% AUC and 92% f1 score with no overfitting.