CASE STUDY

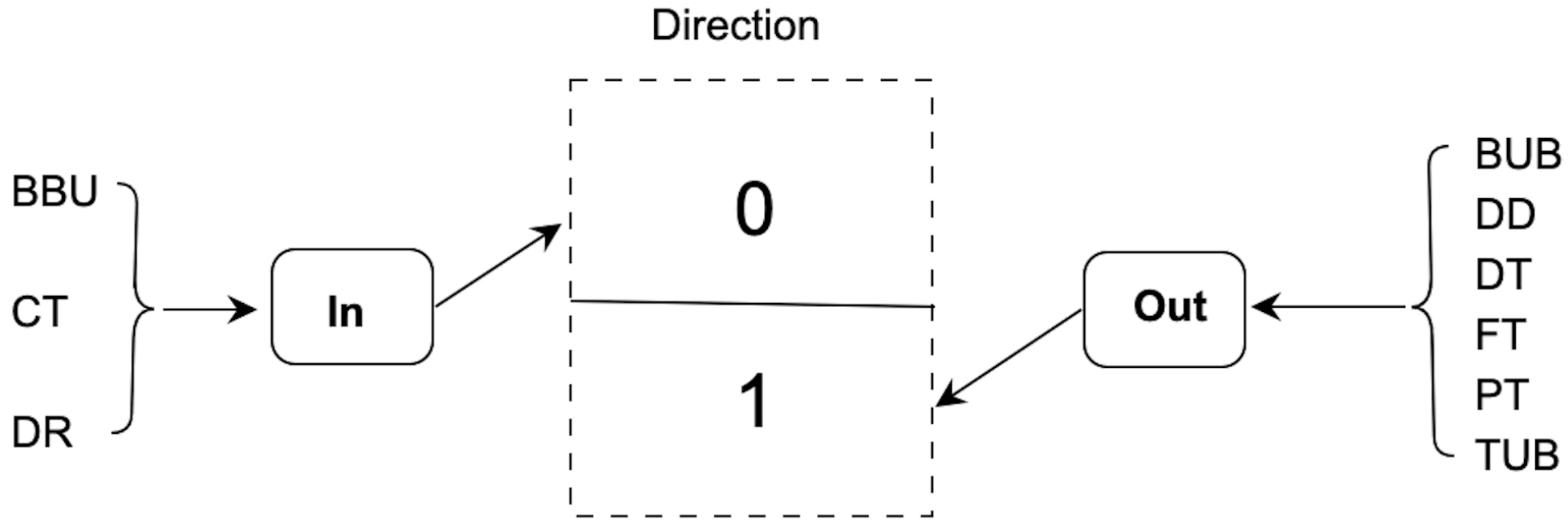
ABDULLAHI A. IBRAHIM

Task

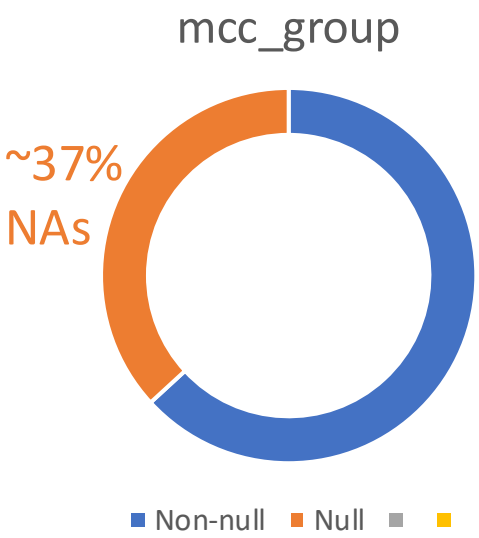
- Predict income and expenses for a holdout sample of ~10k users for the month of August based on a training sample of ~10ks from February – July.
- Based on your judgement of the usefulness of the results, either aggregate the data into incoming & outgoing flows, or predict base on the transaction type/category level

Assumptions

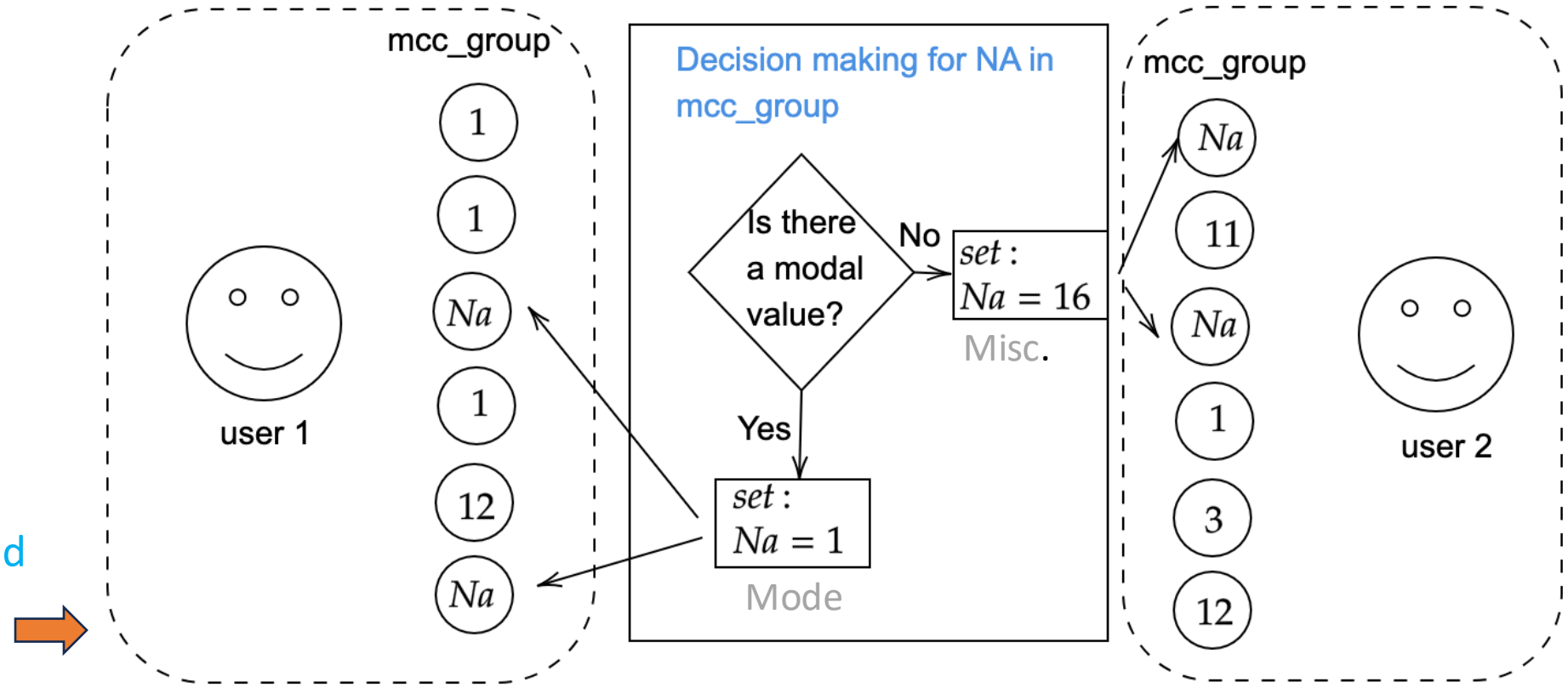
1. Aggregate data into In (incoming) and Out (outgoing) flows.



Fill missing values in mcc_group

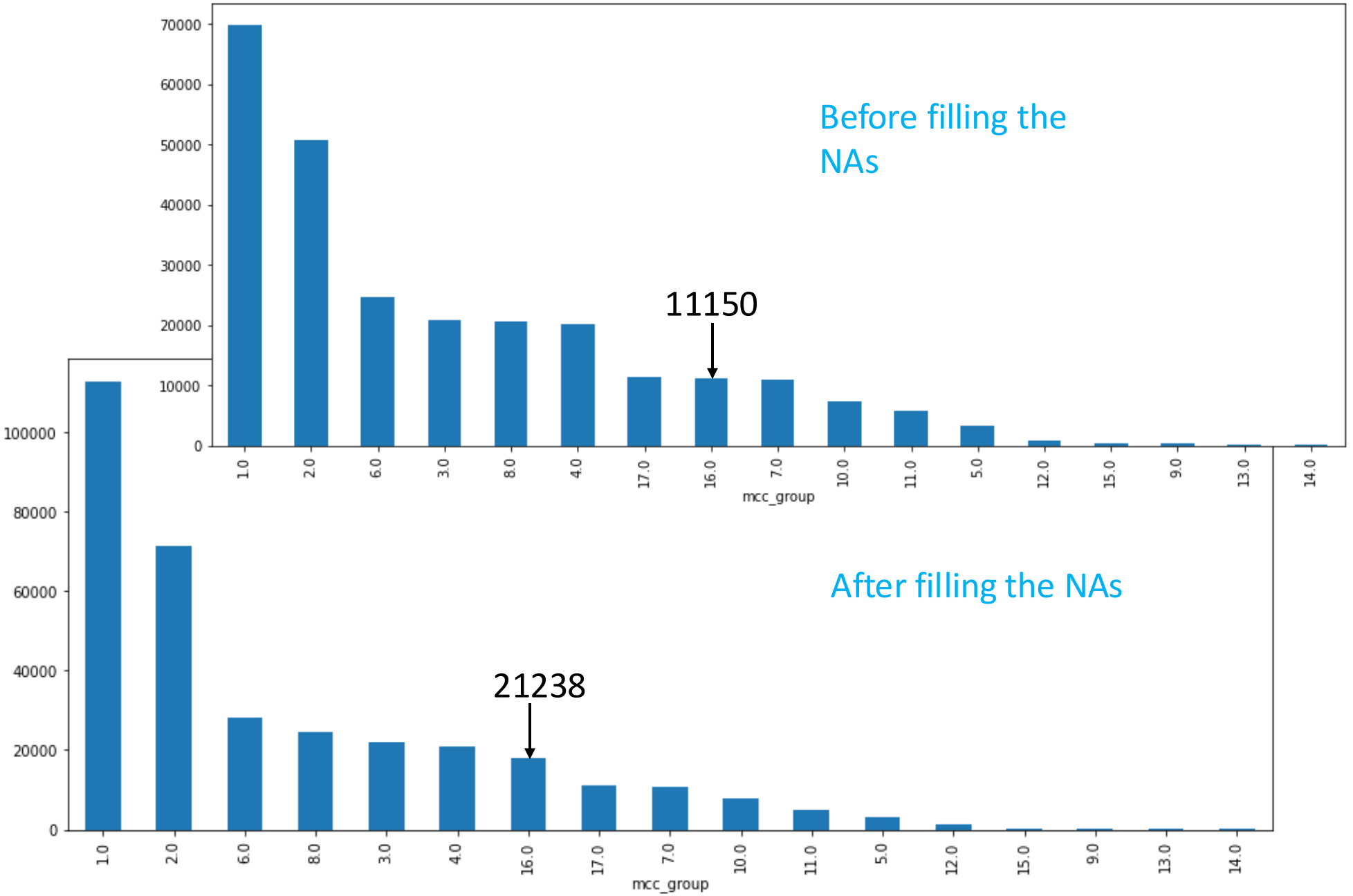


Currently 37% is missing and we fill it as shown here



Impact of filling NAs

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



Feature engineering

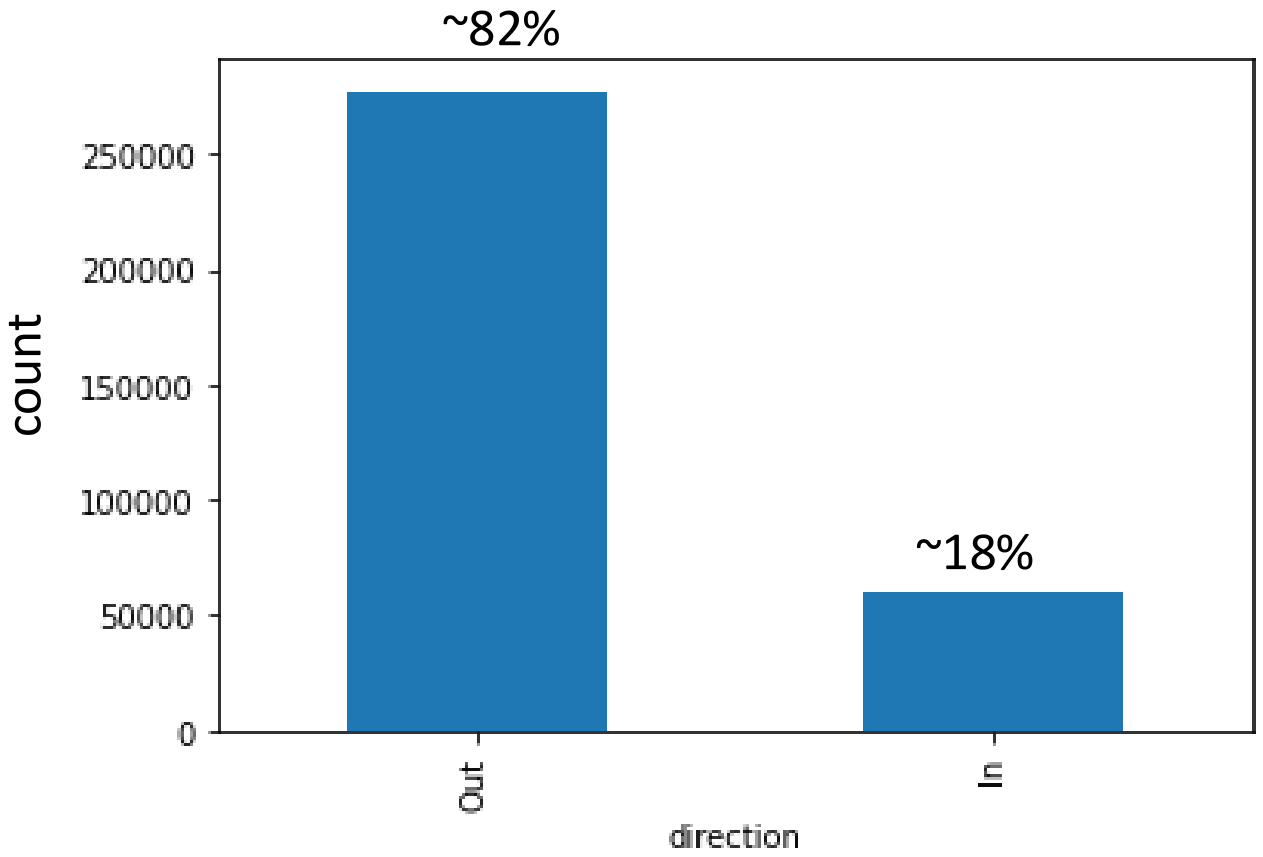
New Features

- Transaction Agent
- Transaction direction
- Transaction month and day

Feature preprocessing

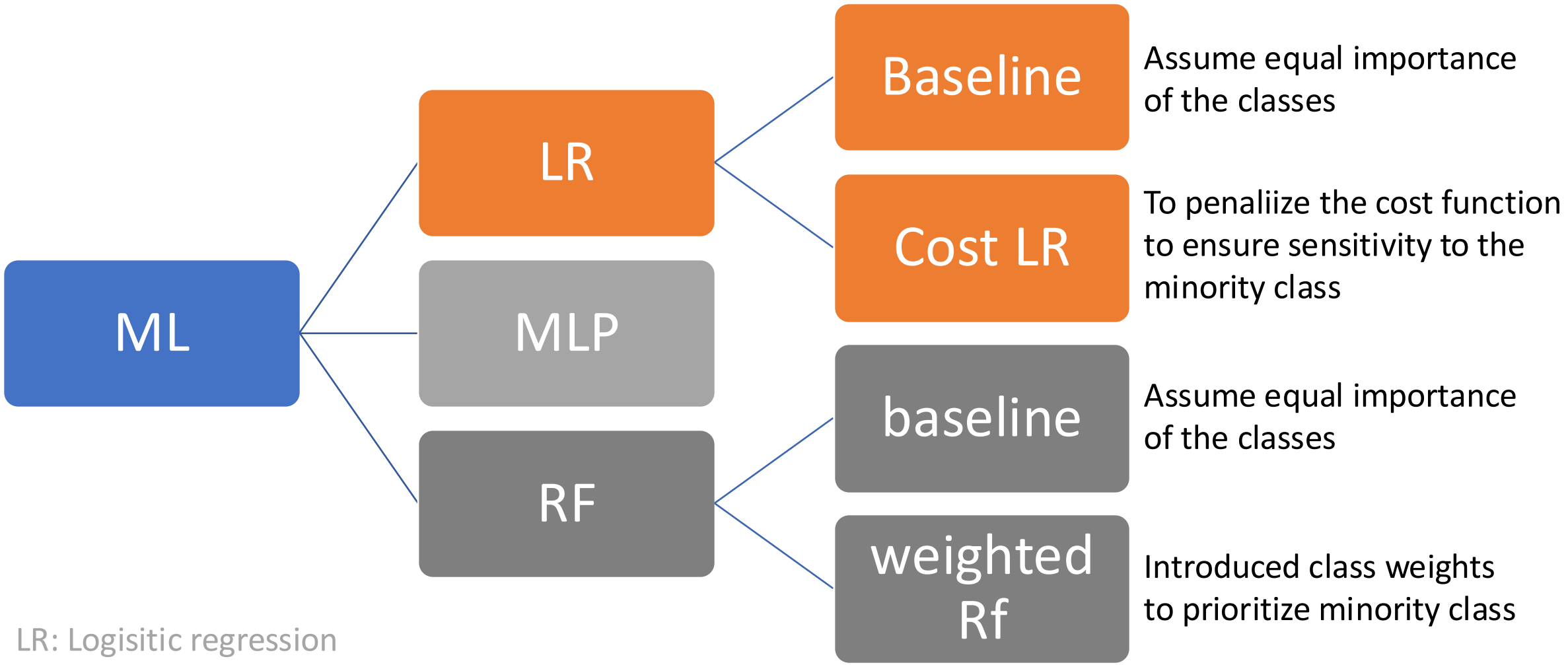
- Encoding categorical features
- Normalization
- Train/valid/test split

Data distribution & split



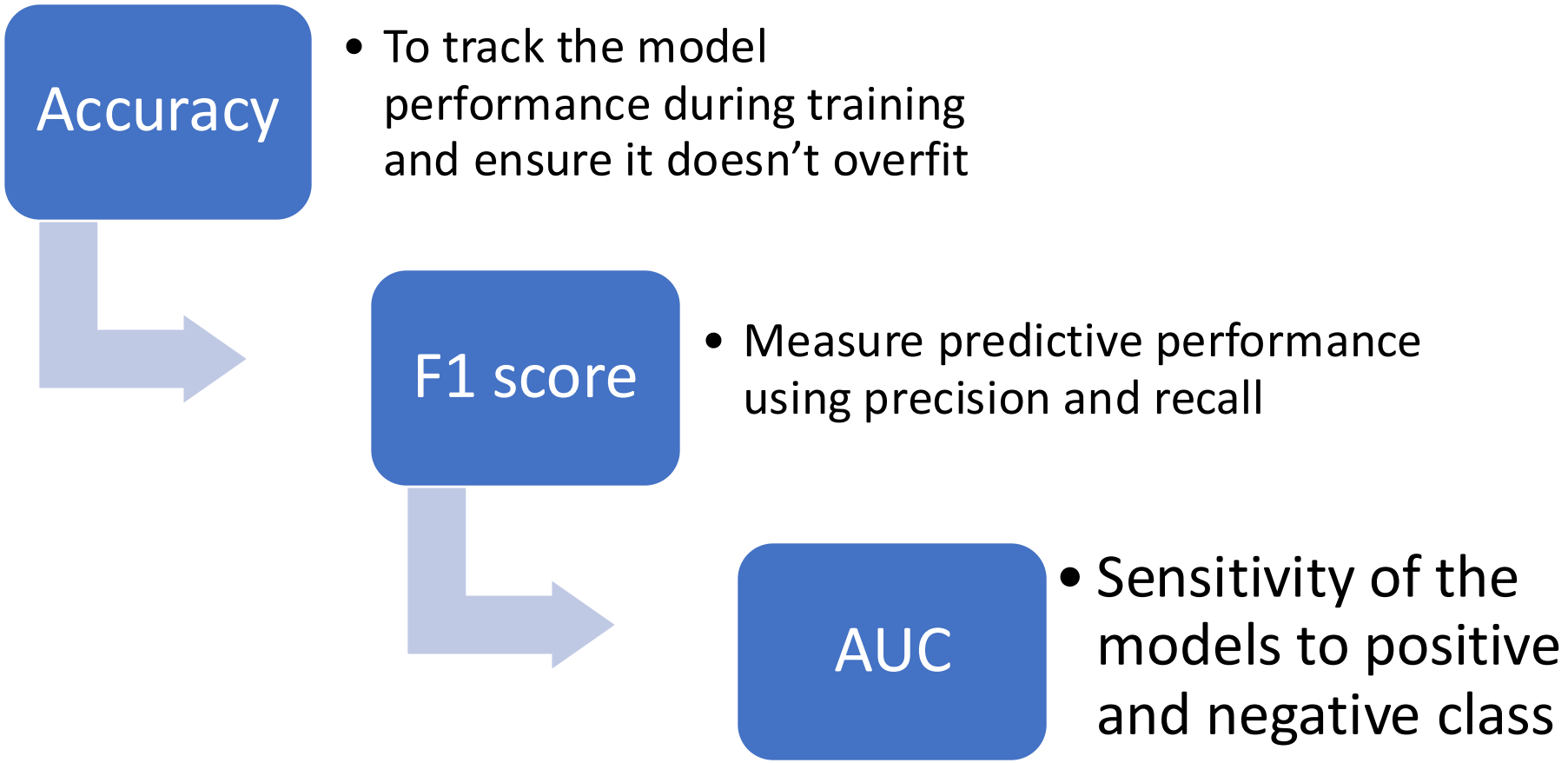
- Shows imbalance distribution
- To avoid skewed result towards Out group, we do **not split randomly**
- Instead, we maintained this proportion across training, validation and test sets

ML methods

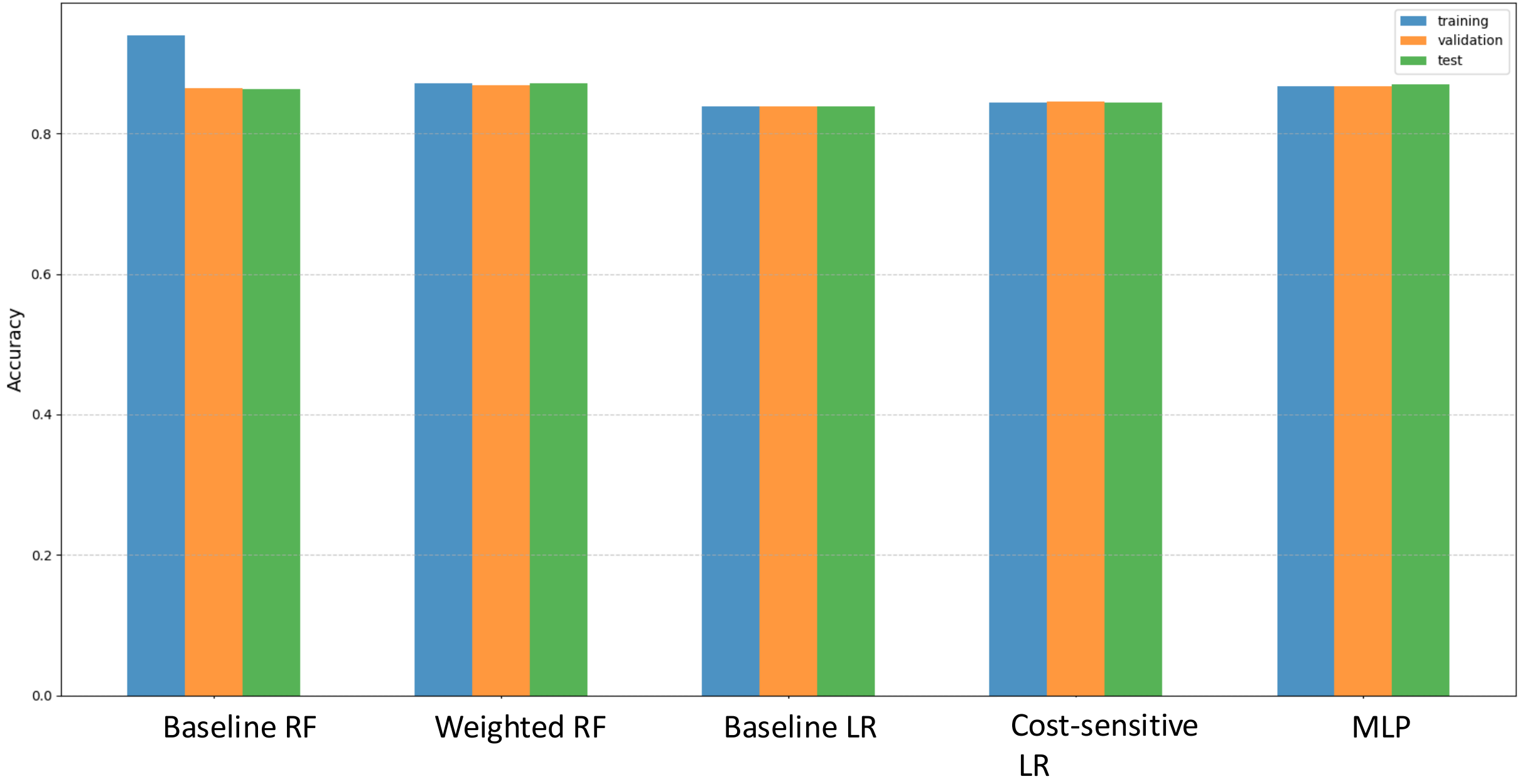


LR: Logisitic regression
RF: Random forest
MLP: Multilayer perceptron

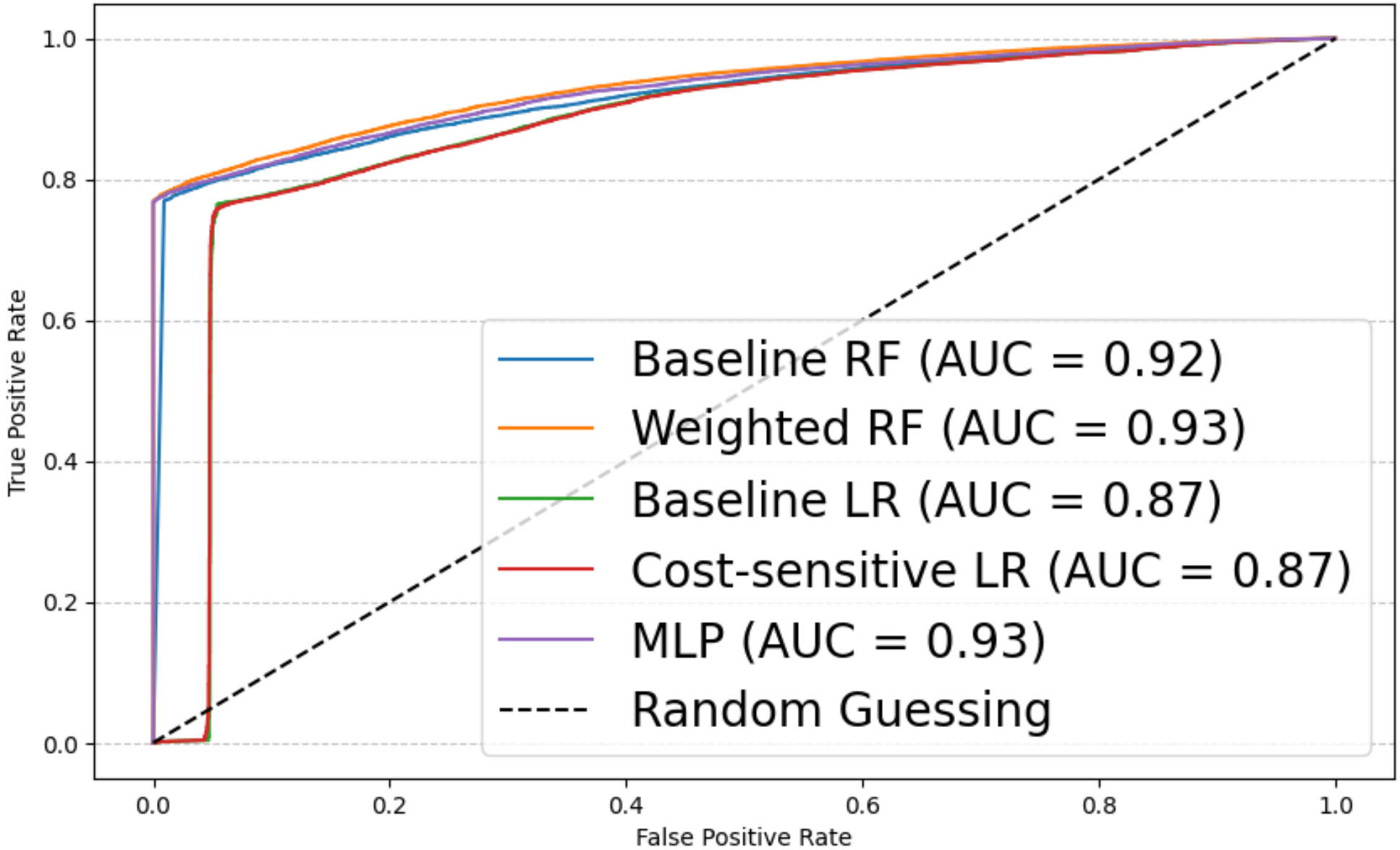
Metrics


Accuracy
• To track the model performance during training and ensure it doesn't overfit
F1 score
• Measure predictive performance using precision and recall
AUC
• Sensitivity of the models to positive and negative class

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.

Thank you