Mobile Banking Fraud Detection

Junyuan Zheng, Ke Cao, Miaochao Wang, Tuo Han
Instructor: Professor Ricardo Collado

STEVENS INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY®
Business Intelligence & Analytics

Introduction

- The intrinsically private nature of financial transactions leads to few publicly available datasets, specially in the emerging mobile money transactions domain.
- The main challenge of fraud prediction is the highly imbalanced distribution between and negative classes.
- Due to the imbalanced nature of fraud detection datasets, we attempt to use under-sampling or over-sampling methods to come up with a suitable approach for real-world problems.

Procedure

- Exploratory Data Analysis (EDA)
- Data cleaning and feature engineering
- Resampling imbalanced training data
- Modeling and model selection
- Results interpretation

Exploratory Data Analysis (EDA)

- Dataset: 6,362,620 mobile money transactions generated by a simulator PaySim[1].
- Highly imbalanced (fraud proportion = 0.12%).

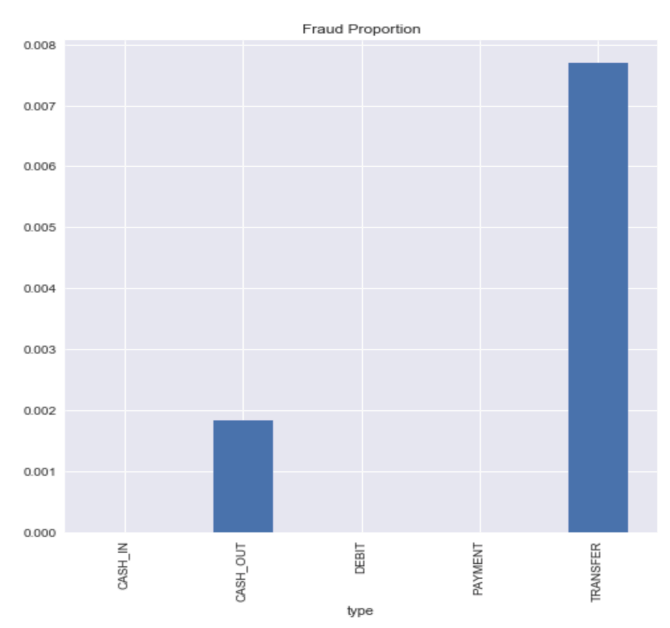
	step	type	amount	nameOrig	oldBalanceOrig	newBalanceOrig	nameDest	oldBalanceDest	newBalanceDest	isFraud	isFlaggedFraud
0	1	PAYMENT	9839.64	C1231006815	170136.0	160296.36	M1979787155	0.0	0.0	0	0
1	1	PAYMENT	1864.28	C1666544295	21249.0	19384.72	M2044282225	0.0	0.0	0	0
2	1	TRANSFER	181.00	C1305486145	181.0	0.00	C553264065	0.0	0.0	1	0
3	1	CASH_OUT	181.00	C840083671	181.0	0.00	C38997010	21182.0	0.0	1	0
4	1	PAYMENT	11668.14	C2048537720	41554.0	29885.86	M1230701703	0.0	0.0	0	0

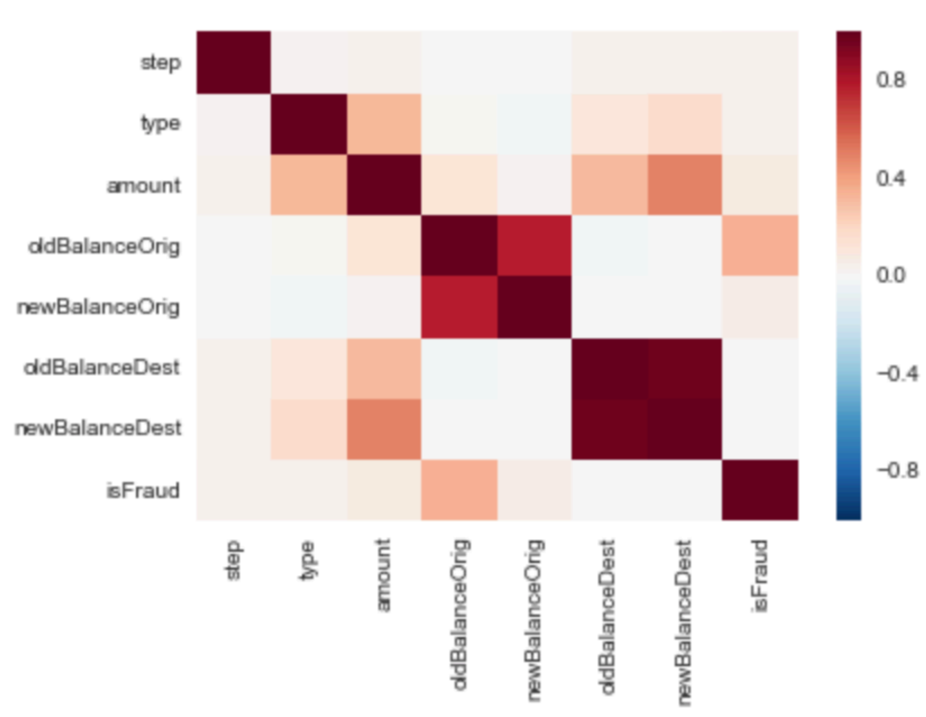
- Fraud only in types of "CASH_OUT" and "TRANSFER"
  → Drop other types and binary encoding feature *type*.

- Account names are not correctly labeled as described ("M" for Merchants, "C" for Customer).
  → Drop *nameOrig* and *nameDest*.

- In some transactions, *newBalanceDest* and *oldBalanceDest* are both 0, while the *amount* is positive and this transaction may not be a fraud. Same situations happen in *newBalanceOrig* and *oldBalanceOrig*.
  → These 0s could be representations of missing values. Replace them by -1.

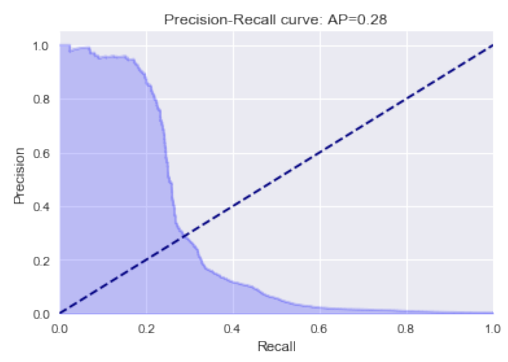
- After data cleaning and feature engineering, all features become numeric without missing values. The correlation heat map is plotted below.

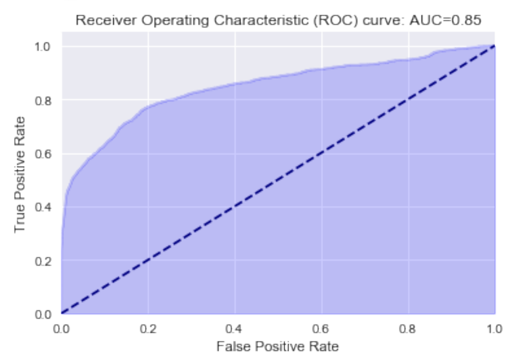




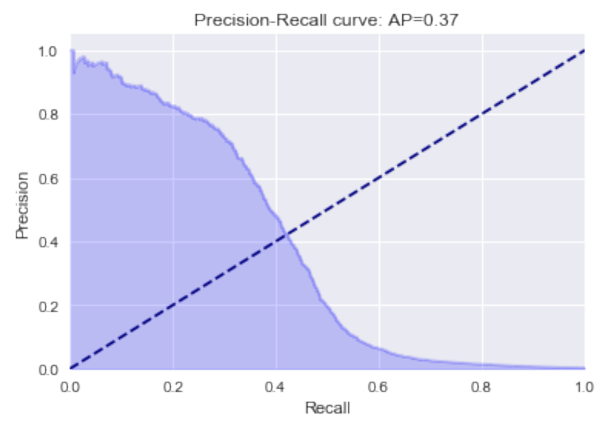
Modeling

- Baseline model: Logistic Regression
- Performance measure: Average Precision (AP)

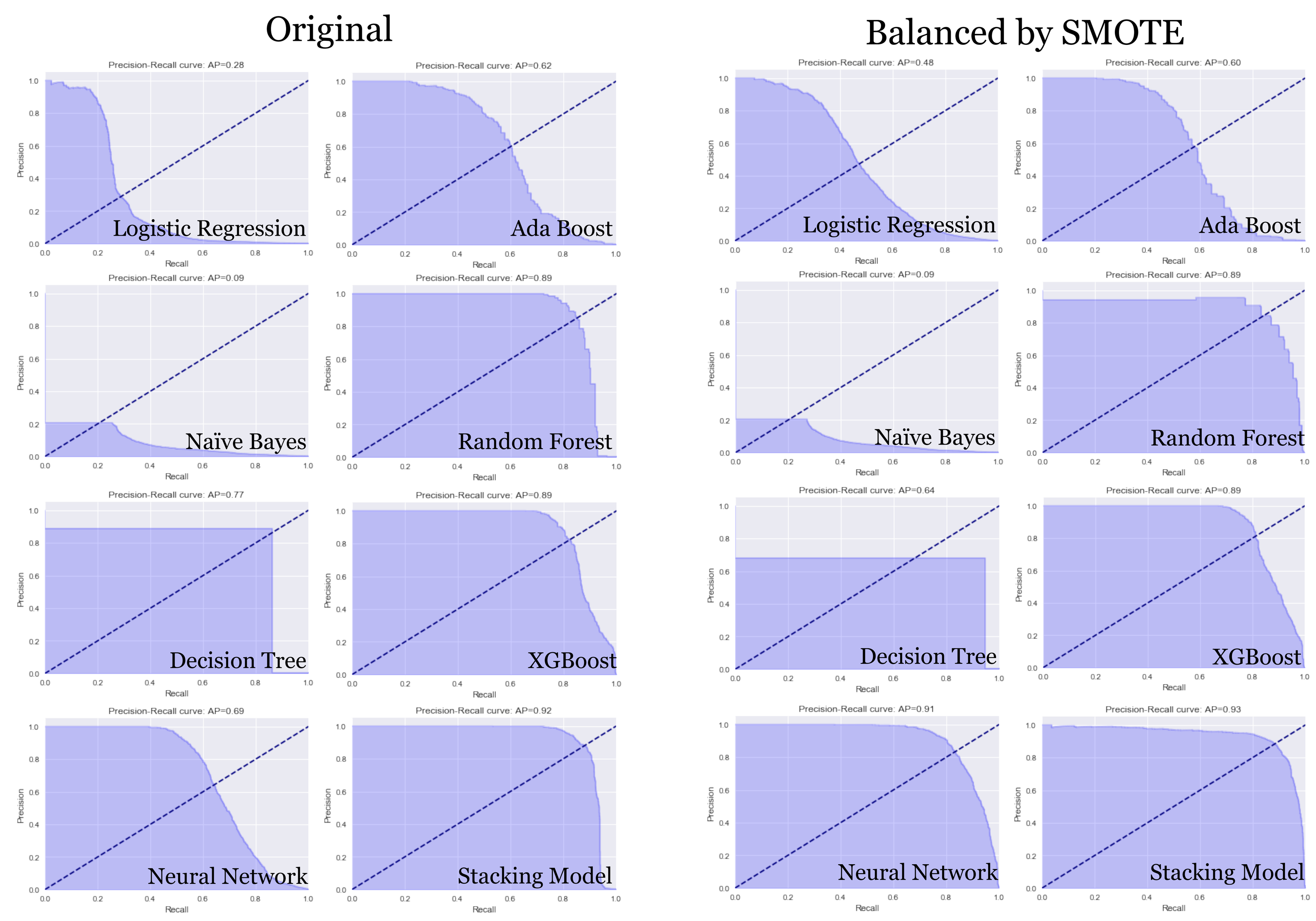




- 300 base models by randomly under-sampling negative class.
- Apply PCA to predicted results and build a stacking model.
  → Time consuming but limited performance increase



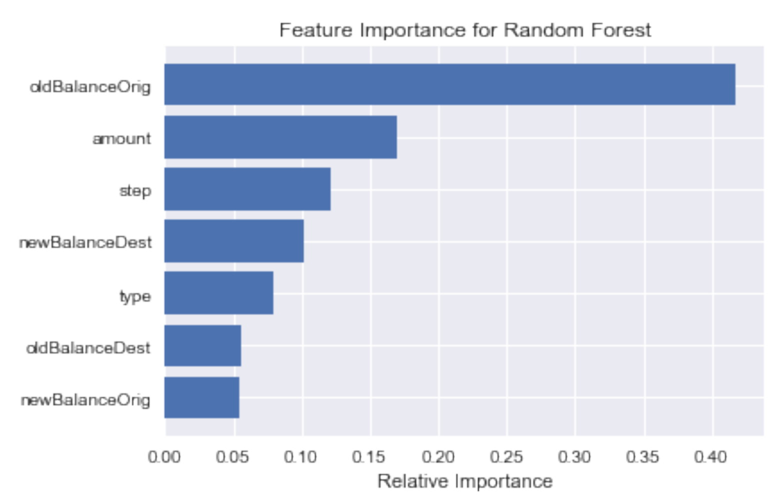
- Synthetic Minority Over-sampling Technique (SMOTE)

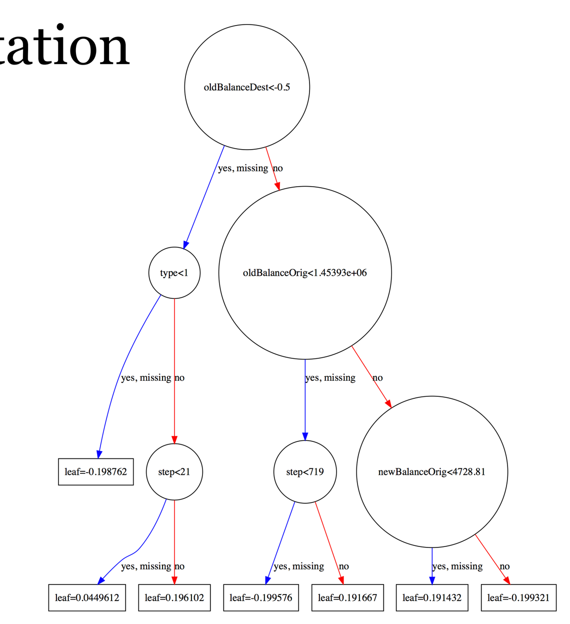


Training set	Precision	Recall	F1-Score	AP
Original	0.89	0.86	0.88	0.92
SMOTE	0.68	0.95	0.79	0.93

Feature importance & tree visualization

- XGBoost is used for the model interpretation





Conclusions

- Imbalanced data decrease the performance of typical machine learning algorithms.
- Ensemble models appear less affected by imbalance.
- SMOTE enhance detection performance by increasing recall.
- Stacking model benefits from diversity of base models.

References

- [1] https://www.kaggle.com/ntnu-testimon/paysim1
- [2] Arjun Joshua, Predicting Fraud in Financial Payment Services https://www.kaggle.com/arjunjoshua/predicting-fraud-in-financial-payment-services/comments
- [3] Ben Gorman, A Kaggle's Guide to Model Stacking in Practice http://blog.kaggle.com/2016/12/27/a-kagglers-guide-to-model-stacking-in-practice/