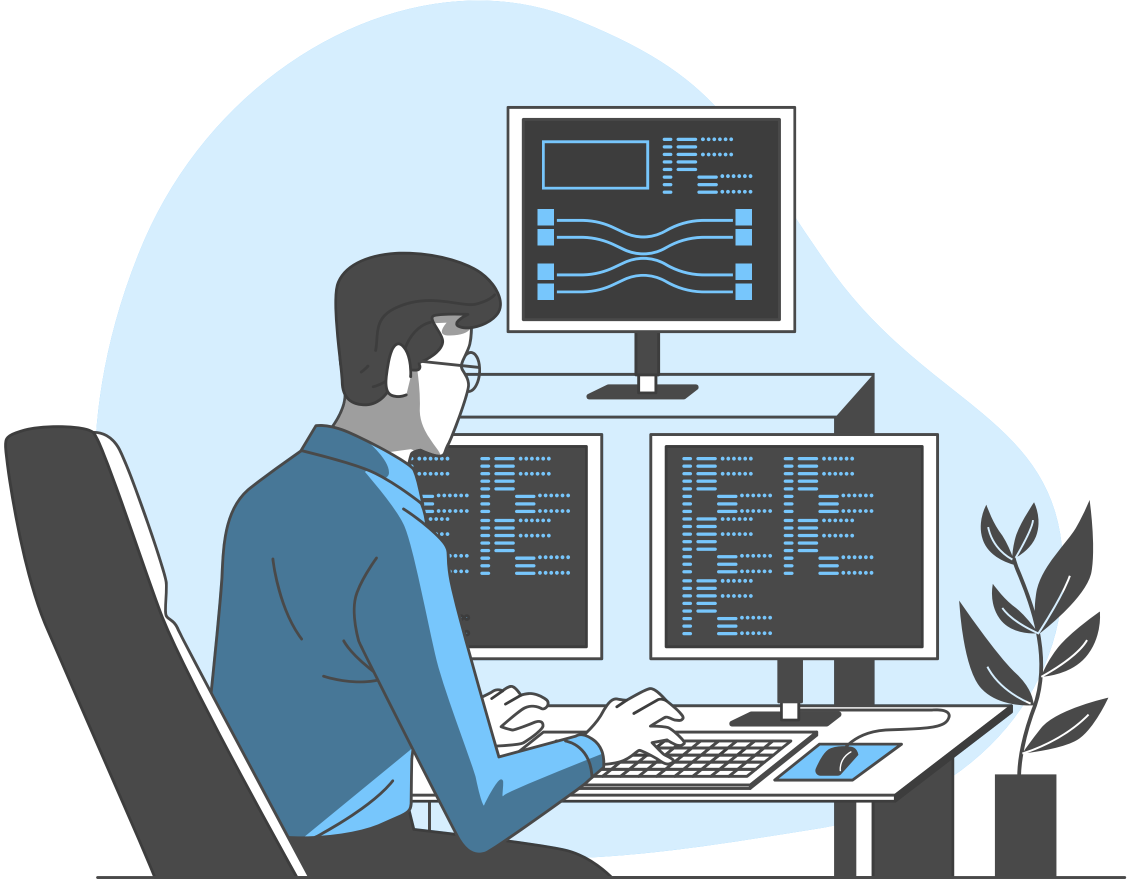


Recent Fraud Analytics

Fraud Team

Company XYZ

The Problem

Identifying and Mitigating Fraud

Payments

Operation member found oddities on 4/27 and flagged payments as fraud

Fraud Risk

Fraudulent payments will lead to decrease in merchant and client trust

Analysis

Mitigate risk by adjusting model and monitoring certain payments

Dataset
4/26 ~ 4/27

1000+

transactions made per day

71%

of users access from iPhone OS

165,000

JPY, largest payment of the period

Fraud Payments Flagged by Operator

Beginner's Un-Luck

First time purchases at respective merchant

Feeling Blue

All purchased from Blue Shop

Window Shopping

All used our app from a Windows NT device

You've Got G-mail..?

No accounts were made using a Gmail account

Time Crisis

Payment timestamps were created before account creation timestamps

A Little Phone-y

70% do not have matching buyer & consumer phone numbers

Filling the Gap

96.9

%

Fraud flag

2,122 missing values will be filled with 0, since these payments were not flagged by the operator

23.4%

Consumer gender

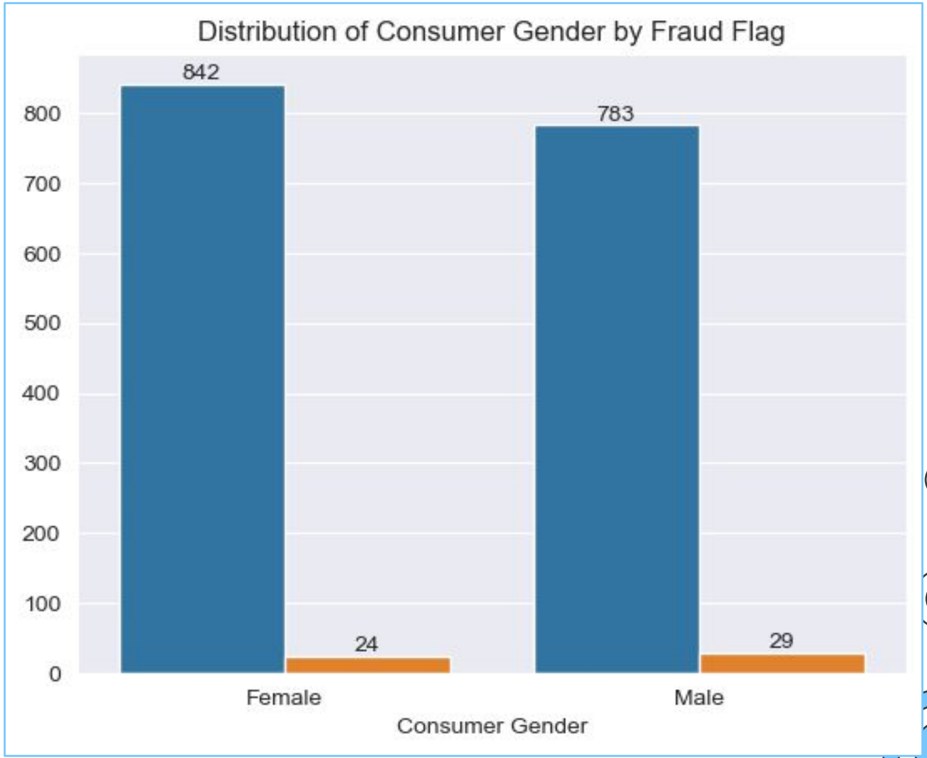
513 missing values, feature will be dropped since distribution is even between fraud and non-fraud

Filling the Gap

23.4%

Consumer gender

513 missing values, feature will be dropped since distribution is even between fraud and non-fraud



Important Features

01 Device used to create the payment

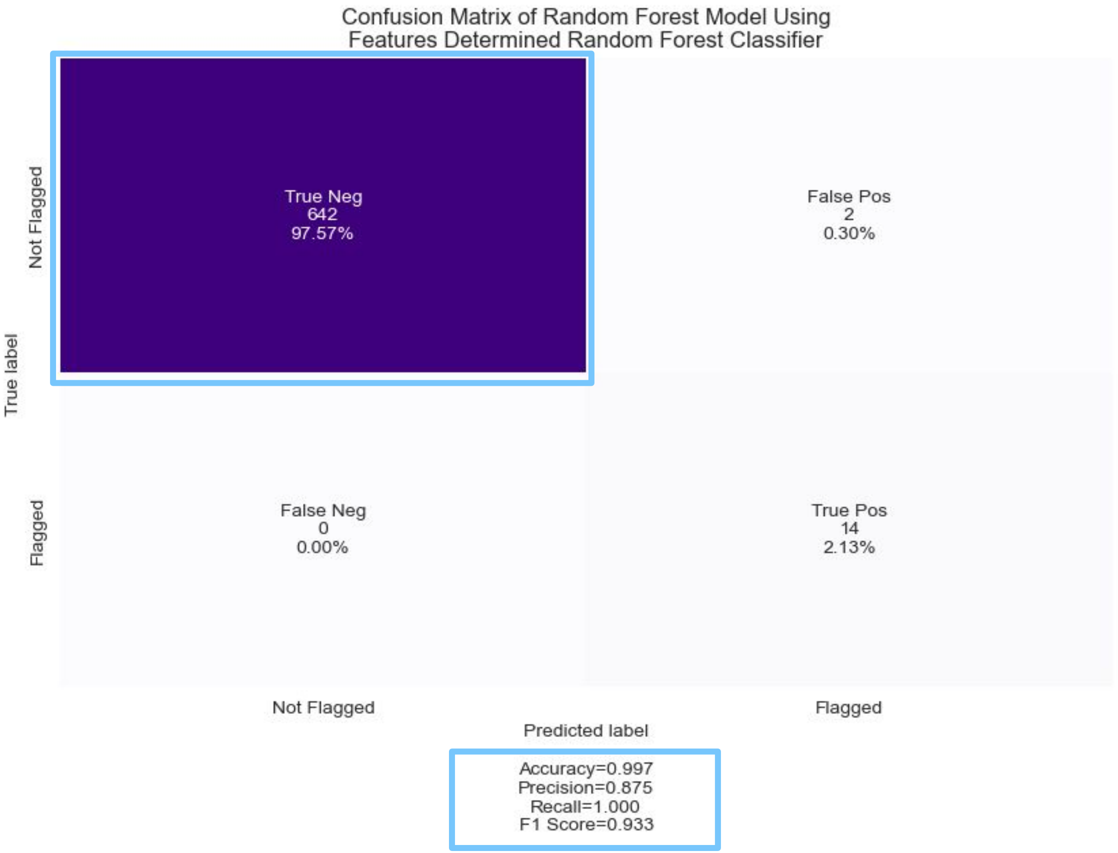
02 Time between payment and account creation

~~03 Device version~~

04 Purchase amount

Result

Model Behavior



Confusion Matrix of Random Forest Model Using Features Determined Random Forest Classifier

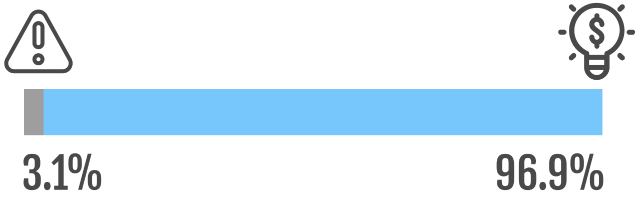
True label \ Predicted label	Not Flagged	Flagged
Not Flagged	True Neg 642 97.57%	False Pos 2 0.30%
Flagged	False Neg 0 0.00%	True Pos 14 2.13%

Accuracy=0.997
Precision=0.875
Recall=1.000
F1 Score=0.933

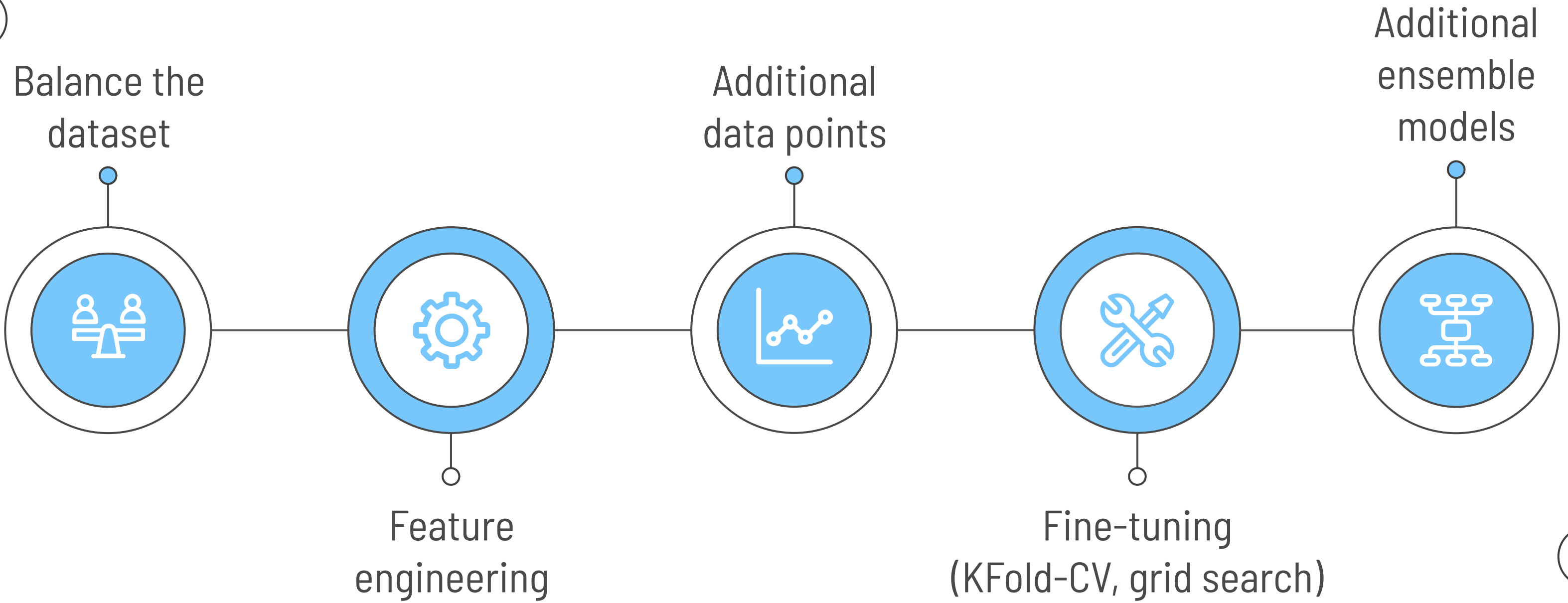
Achieved a **99.7%** accurate model!

Unfortunately, these results are deceiving for multiple reasons:

- 96.9% of payments were not flagged
- Guessing “non-fraudulent” is just as accurate



Future Improvements



Balance the dataset

Feature engineering

Additional data points

Fine-tuning (KFold-CV, grid search)

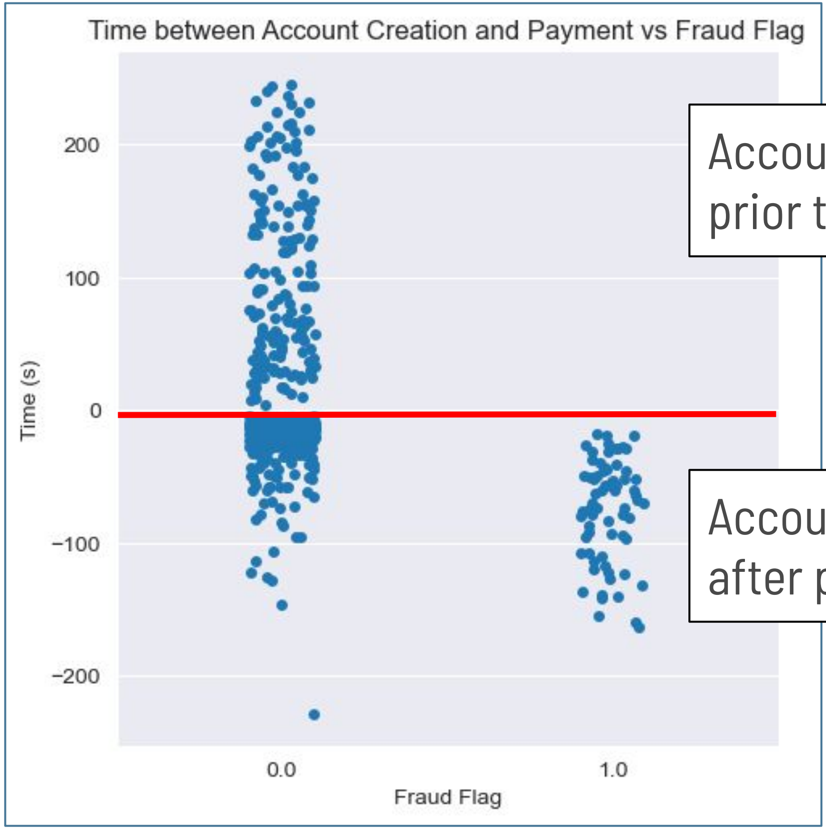
Additional ensemble models

Simple Solutions

Suspicious Time-ing

Monitor payments that occur before an account is created

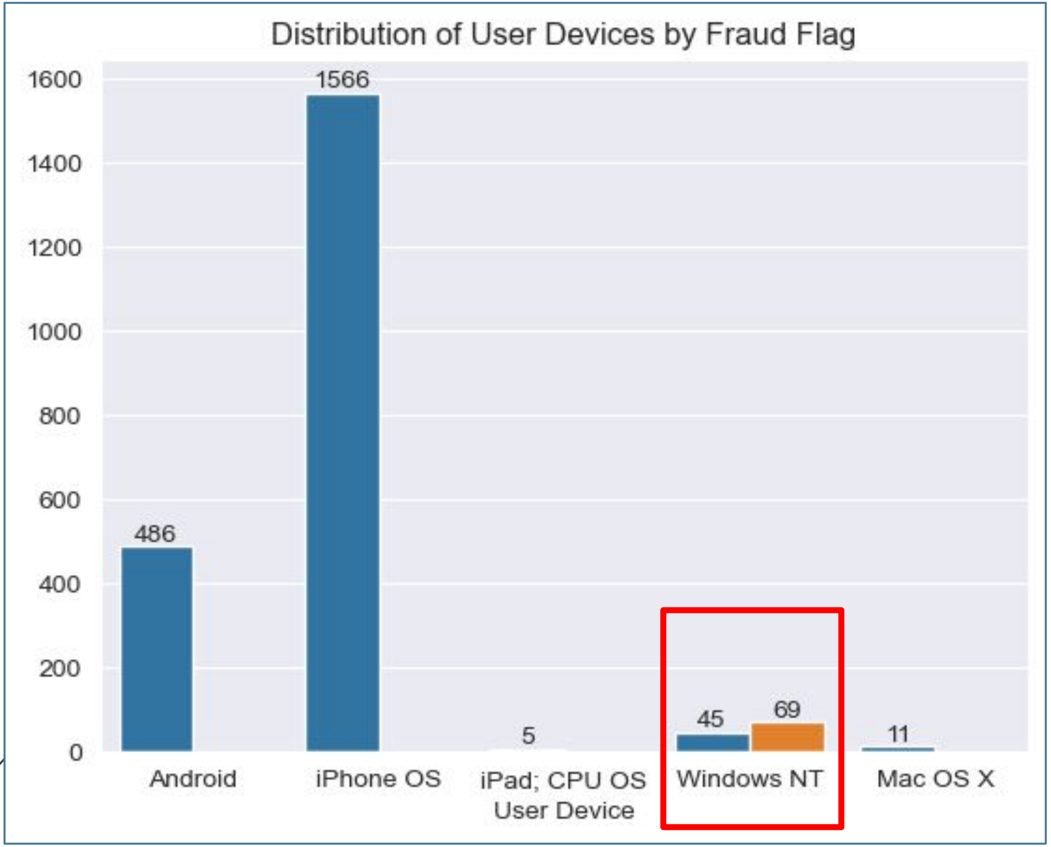
Ask design team how timestamps are created and stored



Account created prior to purchase

Account created after purchase

Windows Pain



Closely monitor users that purchase from a Windows NT device

Observe if suspicious purchases are occurring and block future payments/account

Summing It Up

- Model needs **improving**
- Consider rejecting consumers:
  - purchase from a **Windows NT** device
  - payment made **prior to account creation**

Thanks!

Any questions?

scott@xyz.com
XYZ Company