Feature Selection

Feature selection was important because we had many columns and preferred to keep our models efficient. My XGB had 250 features and would train 6 folds in 10 minutes. Konstantin will need to say what his models had. We used every trick we knew to select our features:

- forward feature selection (using single or groups of features)
- recursive feature elimination (using single or groups of features)
- permutation importance
- adversarial validation
- correlation analysis
- time consistency
- client consistency
- train/test distribution analysis

One interesting trick called "time consistency" is to train a single model using a single feature (or small group of features) on the first month of train dataset and predict `isFraud` for the last month of train dataset. This evaluates whether a feature by itself is consistent over time. 95% were but we found 5% of columns hurt our models. They had training AUC around 0.60 and validation AUC 0.40. In other words some features found patterns in the present that did not exist in the future. Of course the possible of interactions complicates things but we double checked every test with other tests.

Validation Strategy

We never trusted a single validation strategy so we used lots of validation strategies. Train on first 4 months of train, skip a month, predict last month. We also did train 2, skip 2, predict 2. We did train 1 skip 4 predict 1. We reviewed LB scores (which is just train 6, skip 1, predict 1 and no less valid than other holdouts). We did a CV GroupKFold using month as the group. We also analyzed models by how well they classified known versus unknown clients using our script's UIDs.

For example when training on the first 5 months and predicting the last month, we found that our

- XGB model did best predicting known UIDs with AUC = 0.99723
- LGBM model did best predicting unknown UIDs with AUC = 0.92117
- CAT model did best predicting questionable UIDs with AUC = 0.98834

Questionable UIDs are transactions that our script could not confidently link to other transactions. When we ensembled and/or stacked our models we found that the resultant model excelled in all three categories. It could predict known, unknown, and questionable UIDs forward in time with great