Summary of AMEX Kaggle Competition

Zanbing Dai (203rd/4875)

Feature engineering:

1. To handle numerical features that have more than 90% missing values, we only kept their ‘last’ values.
2. Low cardinality numerical features, whose unique values are less or equal to 4, we computed ‘last’ and ‘nuique’ (the number of unique values). The idea come from <https://www.kaggle.com/code/illidan7/amex-basic-feature-engineering-1500-features>.
3. Low cardinality numerical features, whose unique values are between 8 and 21, we created ‘last’, ‘nuique’ (the number of unique values), 'min', 'max', 'mean', 'std' features. The idea initially comes from <https://www.kaggle.com/code/illidan7/amex-basic-feature-engineering-1500-features>.
4. For all the remaining numerical features, we produced 'mean', 'std', 'min', 'max', 'last' features.
5. Finally, we generated ‘last’, ‘nuique’ and ‘count’ for categorical features.
6. We constructed a ‘last-mean’ feature, which is defined as a difference between last and mean. This feature captures the latest changes (<https://www.kaggle.com/code/ragnar123/amex-lgbm-dart-cv-0-7977>).
7. We also invented a ‘max-last ratio’ feature, which is defined as max-last/max-min. This feature helps us to understand better about the latest data. If the latest data is max, then it will be zero. If it is min, then it will be one. If a customer only has the one record, then it will be infinity.
8. After pay features (only used in XGBoost model), difference between payments and balance/spendings (<https://www.kaggle.com/code/jiweiliu/rapids-cudf-feature-engineering-xgb>).

Models:

1. Two LightGBM models, dart: We train two light gradient boosting models with different parameters. Model 1 (.pkl files are save in folder ‘LightGBM model1’) has feature fraction of 0.3 and model 2 has 0.25. It turns out model 1 has out-of-fold cv score of 0.798463 and model 2 has out-of-fold cv score of 0.798061.
2. Two XGBoost models: Model 1 comes from a public notebook in Kaggle. The link is given below:

<https://www.kaggle.com/code/roberthatch/xgboost-pyramid-cv-0-7968>

Model 2 included after-pay features and obtained a cv score of roughly 0.7956.

Ensemble: We submitted two files at the end of competition. We applied two different ensemble techniques.

1. Weighted average (see notebook ‘Weighted average’): 2 lightgbm models + 1 public xgb model. The algorithm is written to basically optimize out-of-fold cv score. Eventually, this model has oof cv score: 0.79956371, public: 0.79928, private: 0.80717.
2. Stacking (see notebook ‘Stacking’): 2 lightgbm models + 1 public xgb model + 1 private xgb model. Fit a logistic regression model to the out-of-fold predictions of these models. This model has oof cv score: 0.79933, public: 0.79939, private: 0.80739.