“Featured Prediction Competition of Home Credit Default Risk”

Xinyi Wei; Liangyawei Kuang

The Hong Kong University of Science and Technology

Presentator: Liangyawei Kuang

Introduction

Background: Many people struggle to get loans due to insufficient or non-existent credit histories. And, unfortunately, this population is often taken advantage of by untrustworthy lenders.

Home Credit strives to broaden financial inclusion for the unbanked population by providing a positive and safe borrowing experience. In order to make sure this underserved population has a positive loan experience, Home Credit makes use of a variety of alternative data--including telco and transactional information--to predict their clients' repayment abilities.

Data

There are 7 different sources of data:

1. application_train/application_test

2. bureau

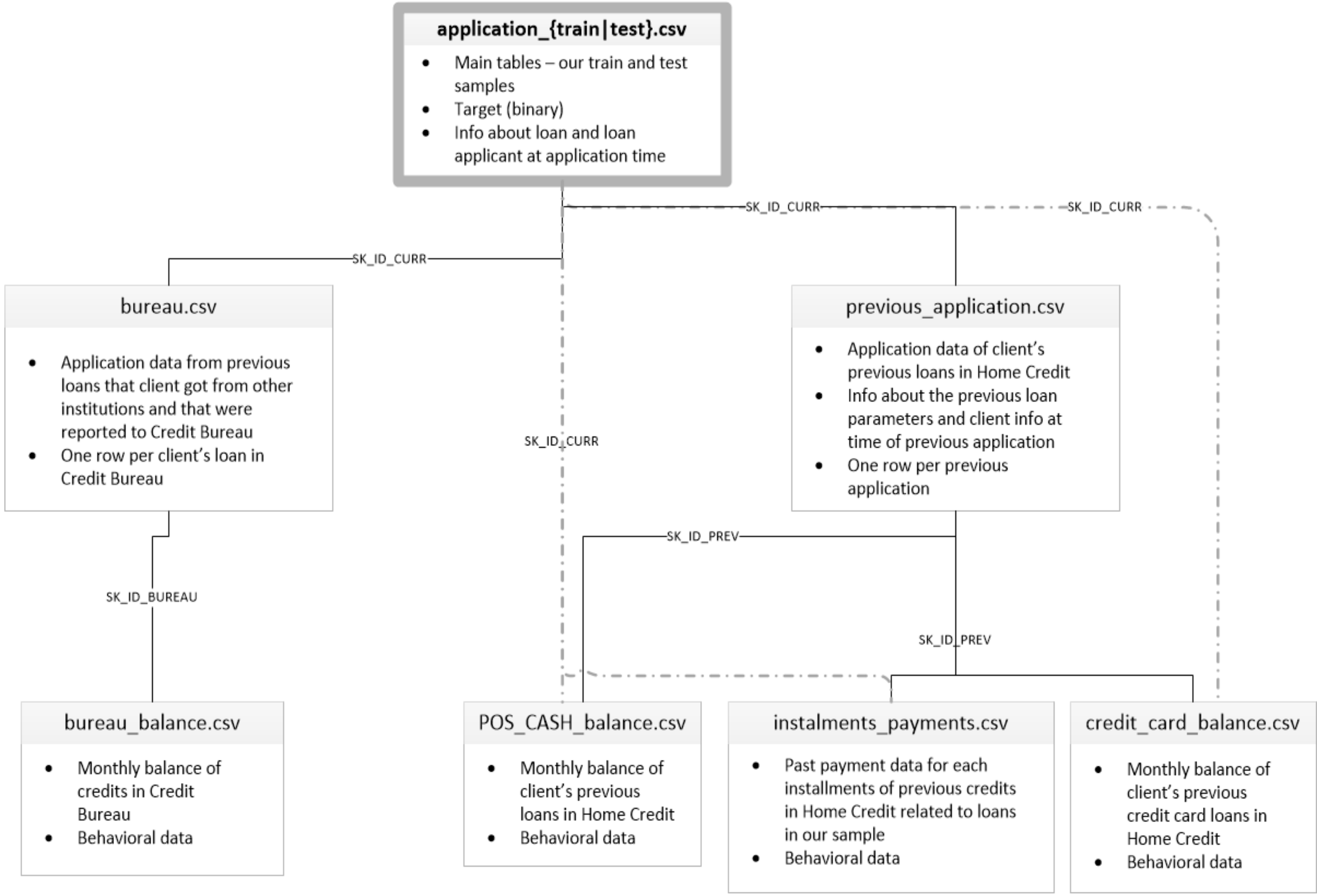
3. bureau_balance

4. previous_application

5. POS_CASH_BALANCE

6. credit_card_balance

7. installments_payment



application_{train|test}.csv

- Main tables – our train and test samples
- Target (binary)
- Info about loan and loan applicant at application time

bureau.csv

- Application data from previous loans that client got from other institutions and that were reported to Credit Bureau
- One row per client's loan in Credit Bureau

previous_application.csv

- Application data of client's previous loans in Home Credit
- Info about the previous loan parameters and client info at time of previous application
- One row per previous application

bureau_balance.csv

- Monthly balance of credits in Credit Bureau
- Behavioral data

POS_CASH_balance.csv

- Monthly balance of client's previous loans in Home Credit
- Behavioral data

instaments_payments.csv

- Past payment data for each installments of previous credits in Home Credit related to loans in our sample
- Behavioral data

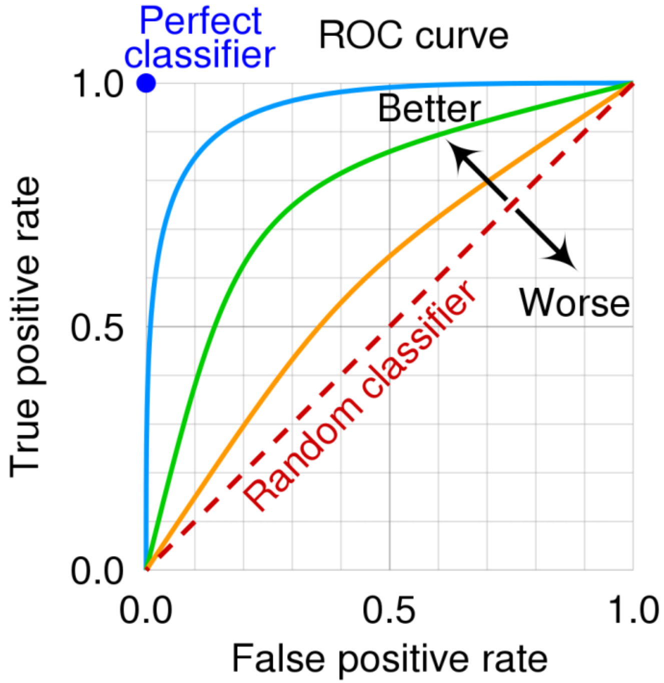
credit_card_balance.csv

- Monthly balance of client's previous credit card loans in Home Credit
- Behavioral data

Evaluation

Title: Home Credit Default Risk
Subtitle: Can you predict how capable each applicant is of repaying a loan?

Submissions are evaluated on area under the receiver operating characteristic curve (ROC) curve between the predicted probability and the observed target.



Constraints

1. No strict latency limits

The latency could be ignored even the models may take a period to make a prediction.

2. High cost for errors

The fault tolerance should be low as an error on missing a potential defaulter could cost huge financial losses.

3. Interpretability

It is hard to have a model with high interpretability when it comes to partial important to decide whether someone is a good loan or a defaulter.

Data Processing and Feature Engineering

1. Exploratory Data Analysis -> which feature is important

2. Missing Value

3. Outliers Processing

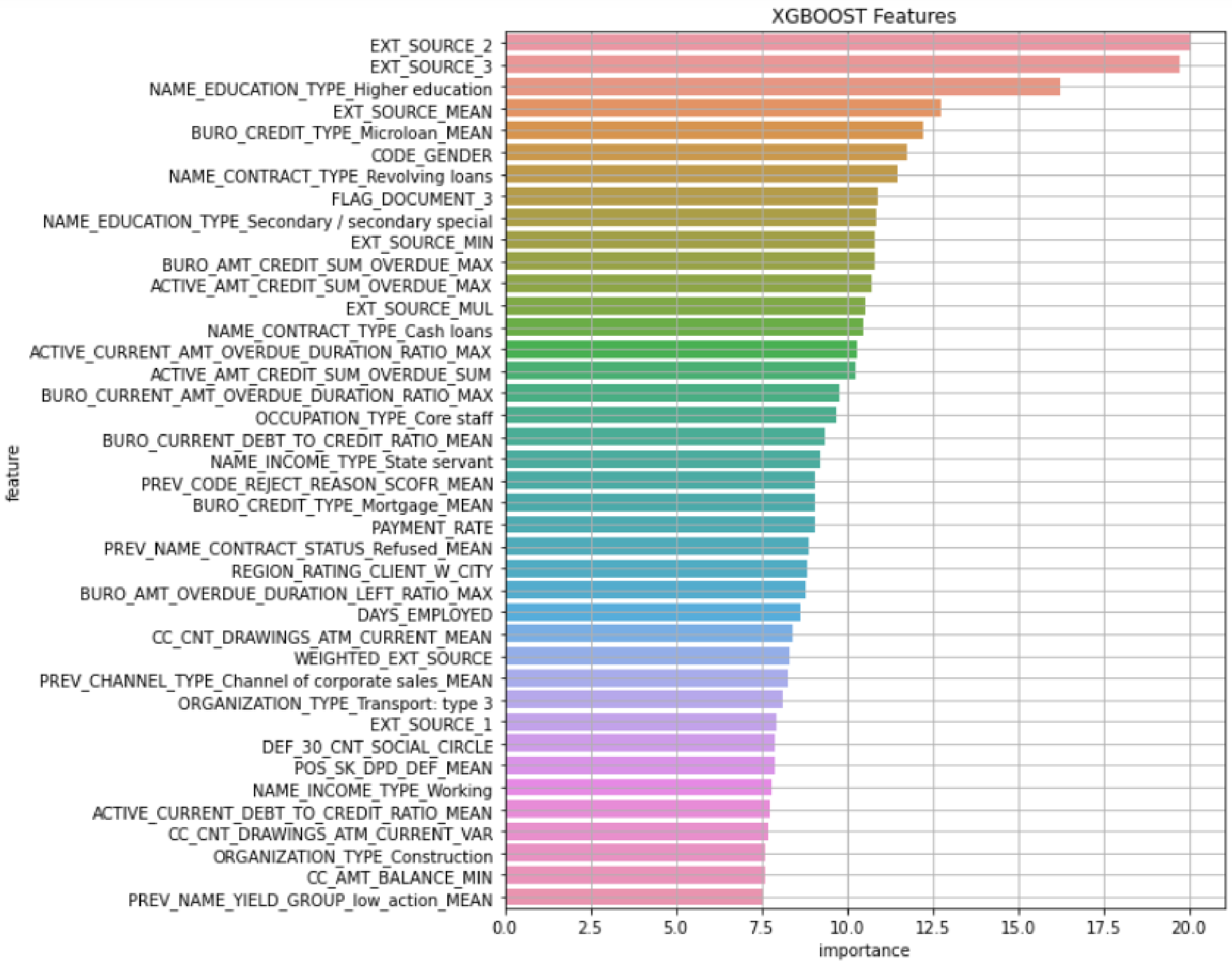
4. Feature Engineering

    1) feature scaling

    2) feature aggregating

    3) categorical features

    4) merge



Basic Methods

1. Stochastic Gradient Descent (SGD) Logistic Regression

   log loss and L2 penalty

2. SGD Support Vector Machine

   hinge loss and L2 penalty

3. Random Forest Classifier

   decision trees, bagging

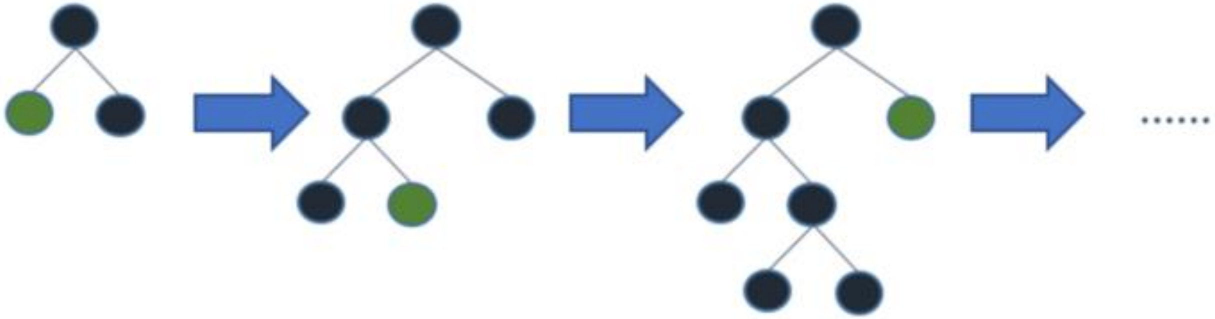
4. Extra Trees Classifier

Tip: The main difference between Random Forest and Extra Trees is that Extra Tree focus on random values for Information Gain during deviding the data for some numeric features.

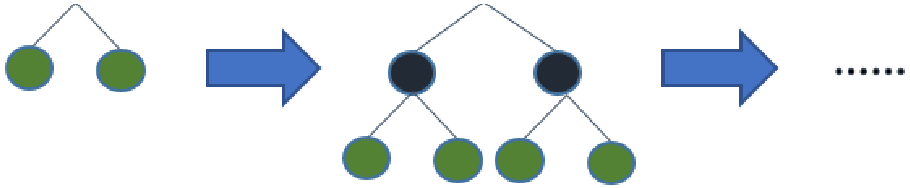
Methods

1. LightGBM Classifier

2. XGBoost Classifier



Leaf-wise tree growth



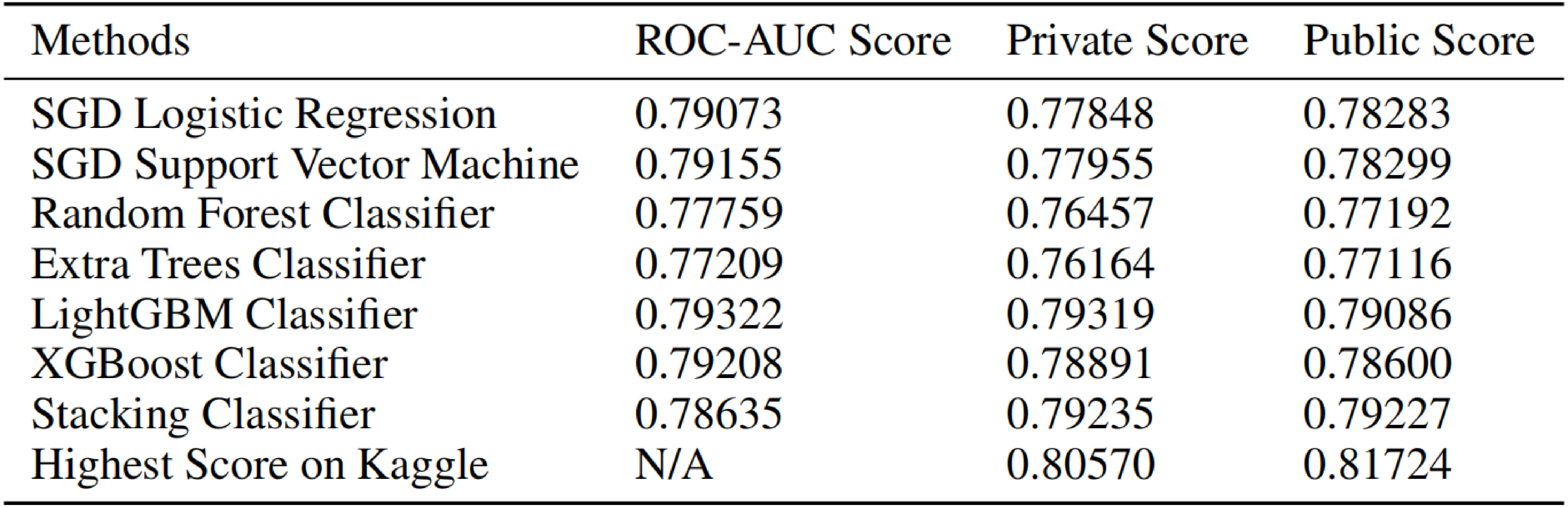
Level-wise tree growth

3. Stacking Classifier

   Logistic Regression + LightGBM + XGBoost
   With a Bayesian Optimizier as a meta classifier

Results with Comparison

Table 1: Comparison among different methods



Methods	ROC-AUC Score	Private Score	Public Score
SGD Logistic Regression	0.79073	0.77848	0.78283
SGD Support Vector Machine	0.79155	0.77955	0.78299
Random Forest Classifier	0.77759	0.76457	0.77192
Extra Trees Classifier	0.77209	0.76164	0.77116
LightGBM Classifier	0.79322	0.79319	0.79086
XGBoost Classifier	0.79208	0.78891	0.78600
Stacking Classifier	0.78635	0.79235	0.79227
Highest Score on Kaggle	N/A	0.80570	0.81724

Existing Solutions

1. 1st Place Solution

   This winner's solution highlighted the importance of feature engineering, which could be proved more useful than tuning and stacking.

2. 2nd Place Solution

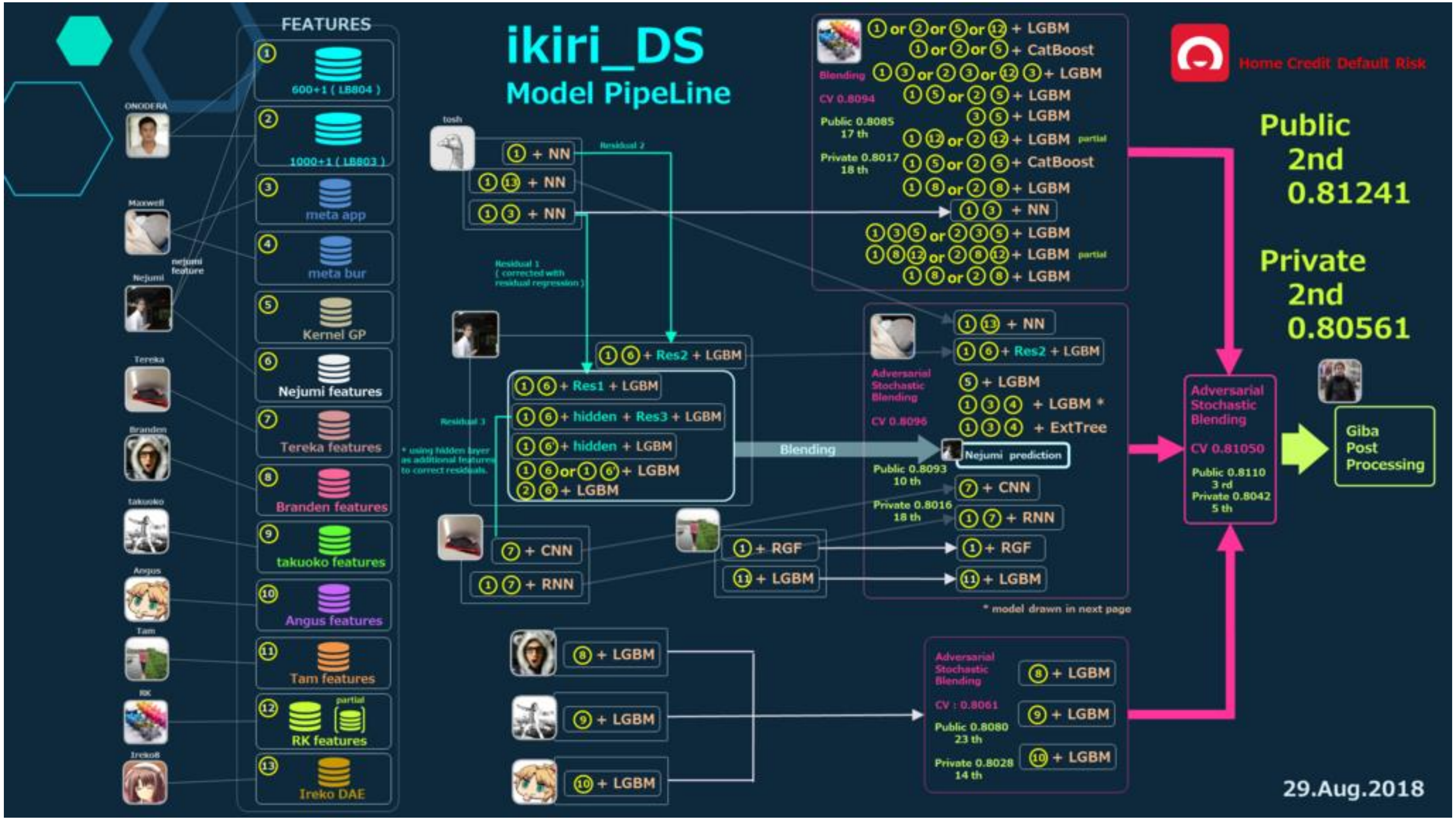
   This team has a complex work arrangements. Except for feature engineering, they also use some end to end CNN and RNN to get time series features from tables.

   For feature engineering, they used various methods, including PCA, UMAP, and T-SNE to reduce the dimensions.They implied LightGBM (with dart), Carboostm DAE for modeling, training and verification. They also applied blending (*Not K-fold for CV, but Holdout set*) methods in the end.

3. 10th Place Solution

   They used the past several months' data seperately and aggregated over current ID for aggregations and implemented LightGBM for feature selection. They also experienced stacking method in the end.

Existing Solutions



Teamwork of the 2nd Place Solution

Conclusions

1. Feature engineering is important when making difficult predictions. In this case, have some expert preknowledge could be more important than having great complex model as these preknowledge could help dive deeper on data set.

2. For stacking methods, we did not implement further with more diverse basic classifiers due to time and human source limits. If we trained different sets of features based on more classifiers, we may have much better results.

Extra

“Coming up with features is difficult, time-consuming, requires expert knowledge. ‘Applied machine learning’ is basically feature engineering.”

—Andrew Ng—