A brief on the approach, which you used to solve the problem.

We created three datasets of more than 1000 features each, built models on each of the dataset, stacked the predictions with Logistic as the meta model. Having a reliable CV strategy was the key to winning the competition.

Which data-preprocessing / feature engineering ideas really worked? How did you discover them?

- The dataset comprises heavily of categorical features which were always on top in the feature importance list. We decided to use that and tried grouping on number of categorical features and created new set of features from numerical variables using multiple aggregation method.
- We created features by interacting multiple categorical columns with another and then creating group features using them, same as above.
- Features such as day, month, day of week, etc were created from Date Features, and more group features were created using them, same as above. The intuition was to create feature that might help models capture the seasonality(as the time period for which the dataset was provided included multiple holidays such as Diwali, Dussehra) or time related information present in the datasets.
- We also created features by interacting multiple numerical features. Interaction we used were primarily division, subtraction between features.
- Features based on nearest neighbors proved out to be very important.

How does your final model look like? How did you reach it?

- Validation Strategy : Group K Fold on DisbursalDate
- Datasets: D1,D2,D3 (three different datasets were used with over 1k features)
- Level 1 Models : LightGBM with different leaves (16,48,128), XGBoost and Catboost on the three datasets
- Level 2 Model: Logistic Regression

What are the key takeaways from the challenge, if any?

- More focus was given on feature creation and building a good validation strategy rather than tuning models.

According to you, what are the 5 things a participant must focus on while solving such problems?

- Understanding the problem statement and the dataset.
- Spend some time on EDA and get a sense of features you are working with by studying the feature importances using any baseline model on initial dataset. This will help in understanding the data and creating powerful features.
- Read papers, blogs, state of the art approaches and kaggle kernels related to the domain of problem statement.
- Build a reliable CV strategy like GroupKFold on DisbursalDate which worked for us in this case.
- And again always keep trying, keep thinking.