APPROACH:

Data:
Train consisted of 54808 rows and test had 23490 rows. Raw dataset contained 12 features.

For modelling purpose we used four variants of the dataset:

- Raw Data i.e. with 12 columns as input
- Modified Data with feature engineered from approach A*, input had 47 features
- Modified Data with feature engineered from approach B*, input had 39 features
- Modified Data with feature engineered from approach C*, input had 32 features

Models Used:

- XGBOOST (version-- 0.72)
- LIGHTGBM (version-- 2.1.1/ 2.1.0)
- CATBOOST (version-- 0.9.1.1)

Validation Scheme:
5 Fold, Stratified scheme was used with random seed as 100 and 0 (for some models)

Hyper-parameter Finding:

- Used Hyperopt (https://github.com/hyperopt/hyperopt) for hyperparameter optimization
- We used both ROC and F1 (based on certain thresholds) to fine tune the hyperparameters

Missing column imputation:

Missing value imputation was done only for catboost. Xgboost and lightgbm can deal with missing values on their own. Missing categorical columns were imputed using a missing category or the mode of that categorical variable depending upon the distribution of data in that variable, while numerical columns were imputed using the mean/median of the numerical variable.

Categorical Encodings:

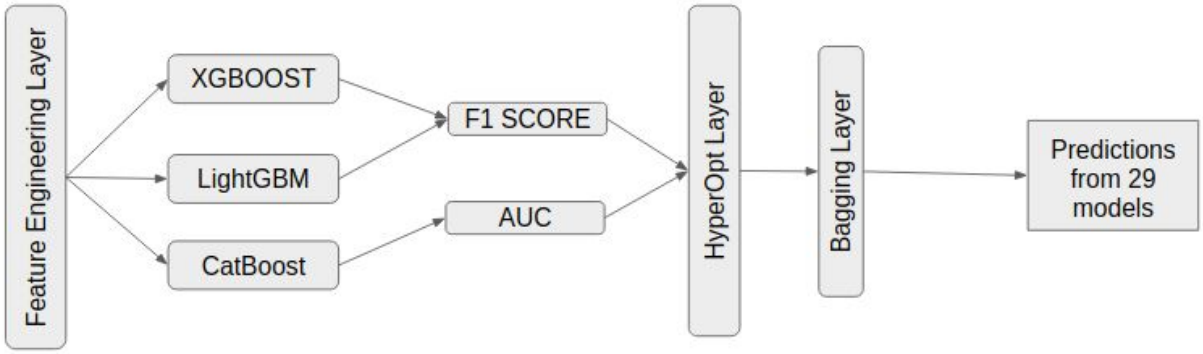
One hot encoding, frequency encoding and target encoding were tried on categorical variables. Frequency encoding was outperforming OHE and TE in the CV scores. So, frequency encoding was the one which was implemented for our three datasets.

Feature Engineering:

The raw dataset had 12 features. New variables were created by grouping by on a categorical variable and calculating the mean of a numerical variable.
On the basis of no of uniques in a particular column, a column was identified as a categorical or a numerical column.
There were three scenarios:

- **Approach A:** 5 categorical variables and 7 numerical variables: This led to addition of 7*5=35 variables which gave a new dataset of 35+12=47 features
- **Approach B:** 9 categorical variables and 3 numerical variables:This led to addition of 9*3=27 variables which gave a new dataset of 27+12=39 features
- **Approach C:** 5 categorical variables and 4 numerical variables (These variables were selected by seeing the feature importance of the lightgbm classifier): This led to addition of 5*4=20 variables which gave a new dataset of 20+12=32 features

FINAL MODEL:



Our final model is an ensemble of 29 models consisting of:

- The optimum threshold was the one which was maximizing the f1 score for the CV predictions. 5 thresholds, 2 lower than the optimum threshold and 2 higher than the optimum threshold and the optimum threshold were chosen to give more robust predictions.
    - 5 catboost models with 5 different thresholds on the raw dataset of 12 features.
    - 5 catboost models with 5 different thresholds on the raw dataset of 39 features(Approach B)
- For a particular xgboost/lightgbm classifier, the optimum thresholds in the 5 folds were varying from 0.25-0.30. After having seen this pattern of the optimum fold wise thresholds(more details mentioned below), 19 models were created using different thresholds and different dataset creation approaches mentioned above.
    - 4 xgboost models on Approach A
    - 5 xgboost models on Approach B

    - 5 lightgbm models on Approach C
    - 5 lightgbm models on Approach A

Approach for deciding the F1 score threshold:
Refer to this paper for details: http://iranarze.ir/wp-content/uploads/2016/10/E2281.pdf

Things that didn't work out

- Target encoding and One Hot Encoding on categorical variables didn't perform as well as frequency encoding.
- Bayesian / Genetic Feature Selection didn't give a boost to the performance metric (F1 score/AUC)
- Combining different numerical features also didn't show a boost in the performance.