EDELWEISS HACKATHON-MACHINE LEARNING

Solution 1 - Predict Forclosure Probability

Author - Anurag Sharma

PROBLEM STATEMENT

- To predict the probability of foreclosure for each of the datapoints in the test set based on the training dataset.
- Training dataset contains Customer demographics, Customer transactions, List of foreclosed customers, and Email interaction of the customer with the customer representative.

TOOLS USED

- Python
- Pandas/Numpy
- Matplotlib
- Sklearn
- XGBoost

SOLUTION - 1

- Read the dataset into pandas dataframe.
- Then I analysed LMS data file which contains customer transactions. It contained transactions for different dates.
- So I sorted the data based on LAST_RECEIPT_DATE and drop the duplicates based on AGREEMENT_ID.
- Then I merged train_forclosure.csv and lms_31JAN.csv files based on AGREEMENT_ID.

SOLUTION - II

- In the next step I did feature engineering.
- I created several new features based on the given features like loan amount difference, tenor difference, ROI difference, total tenure etc.
- These features helped me to increase my score on the leaderboard as they were giving a lot of information about the output variable.
- Apart from these handmade features, I generated few features from datetime variables like Authorization day, month, year, Interest day, month, year etc.

SOLUTION - III

- After feature generation, I checked for null values in the dataset and impute them based on the variable.

- Then I did feature transformation. I transformed some continuous features so that they can follow normal distribution. If our data has a Normal distribution, the parametric methods are powerful and well understood.

- For categorical features like CITY and PRODUCT, I did label encoding to convert them into numeric features.

MODELING

- For model training, I used XGBoost Classifier. XGBoost is an implementation of gradient boosted decision trees designed for speed and performance. XGBoost is an algorithm that has recently been dominating applied machine learning and ML competitions for structured or tabular data.

- I compared XGBoost with different ML algorithms but it outperformed all of them in terms of performance and speed.

- After tuning hyper parameters, I got these as the best on this dataset,

    - num_rounds = 1200

    - learning rate = 0.01

    - max_depth = 8

    - sub_sample = 1.0

    - colsample_bytree = 0.8

RESULTS

- From this solution I got 99.94244 ROC-AUC score on the leaderboard.

“In God we trust, all others must bring data.”

-W. Edwards Deming

THANK YOU