

**Data Understanding**

Presenting a detailed overview of the data.

Important points:

* Size and dimensionality of data.
  + Number of rows in train set is 1106673 rows while Number of rows in test set is
  + There are 62 columns including target columns.
  + The memory usage is 520 mb+ which is very high.
  + To reduce the memory usage, a code block has been added ( memory usage reduced substantially).
* Important summary statistics of the variables (including distribution of target variable)?
  + There are 11 categorical columns, 25 float32 columns and 25 int64 columns.
  + Record\_column is right skewed, while most the numerical columns are evenly distributed.
  + Both Target columns have class imbalance:
    - Primary\_close\_flag has 12% class imbalance.
    - Final\_close\_flag has 19 % class imbalance.
    - To tackle class imbalance, scale\_pos\_weight parameter is used.
* Details of blank / null values and outliers:
  + Number of missing values in train set is 907408.
  + Number of missing values in test set is 389112.
  + Since the percentage of missing values is less than 5%, no imputation or handling of missing values is performed.
  + There are various columns which have outliers shown in EDA.

**Data Preparation**

Presenting how data was cleaned and prepared for building model.

Important points:

* How missing values were handled?
  + Since the percentage of missing values is less than 5%, no imputation or handling of missing values is performed.
* How were the outliers handled?
  + Outliers were not removed from the data.
* Which new features / derived features were created from the data?
* No feature were created.

Was binning required? If yes, why and how was it done?

* Details of data partitioning:
* Data partitioning was done in cross-validation
* Any other important step used by the team:
* Categorical features were converted into numeric representation of feature by identifying distinct values.

**Model Building & Evaluation**

Presenting a detailed overview of model(s) used and the results observed.

**Important points:**

* **Scale Pos Weight Calculation**: The code calculates a weight, **scale\_pos\_weight**, to address class imbalance by computing the ratio of '0' values to '1' values in the target variable **y\_1**.
* **Hyperparameter Setup**: It defines hyperparameters for the CatBoostClassifier, such as the number of estimators, the objective function (Logloss), and the evaluation metric (AUC), while considering the calculated scale\_pos\_weight.
* **Model Initialization**: The CatBoostClassifier model is initialized with the specified hyperparameters.
* **Cross-Validation Configuration**: A 5-fold StratifiedKFold cross-validation setup is created to assess the model's performance across different subsets of the data.
* **Data and Evaluation Arrays**: Arrays are initialized to store predictions, scores, and feature importance values for analysis and evaluation.
* **Cross-Validation Loop**: The code enters a loop to perform k-fold cross-validation, including model training and evaluation for each fold.
* **Results and Final Predictions**: It calculates ROC AUC scores for training and validation sets, prints fold-wise results, calculates mean scores, and computes final test predictions by averaging predictions from all folds.

**Results and Recommendations**

A summary of the important insights and results obtained and recommendations for deployment!

Important points:

* **The developed model is Catboost.**
  + Mostly hyperparameters are kept default.
  + N\_estimators: 5000
  + Scale\_pos\_weight for class imbalance.
  + Eval\_metric as AUC.
* **Categorical to Numerical Conversion**: Perform the conversion of categorical columns into numerical format to ensure compatibility for machine learning algorithms.
* **No Imputation Needed**: Given the less percentage of missing values, there is no necessity for imputing null values within the dataset.
* **Memory Optimization**: Implement memory optimization techniques in the code to efficiently handle large datasets and reduce memory consumption
* **Top 5 Important features are:**
  + Primary\_term
  + Days\_till\_primary\_close
  + Encoded\_loans\_credit\_type
  + Days\_since\_opened
  + Days\_since\_confirmed.