Approach Document:

EDA:

Chart, bar chart

Description automatically generated

When the customer had purchased credit products and his/her vintage is above 60 ,the lead conversion is higher

Chart, box and whisker chart

Description automatically generated

Avg Account balance is highly skewed with a positive tail, Needs some transformation

Chart, histogram

Description automatically generated

Lead conversion rate is generally higher in age groups between 40 and 60

Chart, bar chart

Description automatically generated

Male customers have higher conversion rates than females

A picture containing graphical user interface

Description automatically generated

Figure shows top region codes by conversion rate of leads

Chart

Description automatically generated

Missing values tend to have higher conversion rates as compared to No/Yes for credit products.Hence ,it makes sense not to impute it with yes/no

Chart, bar chart

Description automatically generated

Self employed Employees have higher conversion rate than others which makes sense since for running a business they must take loans

Chart, histogram

Description automatically generated

Channel code X2 has higher conversion rate and ratio of conversion in X2,X3 are higher

Chart

Description automatically generated

Attrition happens higher in the region marked in yellow,All other bins have more or less equal attrition and retention ratio

**Missing value Imputation:**

Missing values are present in credit history and it was imputed with Not available as per the EDA analysis

**Outliers:**

Binning and log transformation was done to handle outliers(skewness)

**Features:**

Avg\_Monthly\_acc\_balance\_log

'Vintage\_boxcox'

'Vintage\_log'

'Avg\_acc\_bal\_per\_month'

'Avg\_Account\_Balance\_sqrt'

'Avg\_acc\_bal\_10\_bins'

'Avg\_acc\_bal\_20\_bins'

'Avg\_acc\_bal\_30\_bins'

'Age\_5\_bins'

'Age\_10\_bins'

'Age\_20\_bins'

**Feature Engineering:**

**Target Based**

Created Target mean encoded features(both absolute and relative conversion rates)

'LeadConvRate\_by\_Vintage'

'LeadConvRate\_by\_Age'

'LeadConvRate\_by\_Age\_X\_bins'

'LeadConvRate\_by\_Region\_Code'

'LeadConvRate\_by\_Occupation'

'LeadConvRate\_by\_Active'

'LeadConvRate\_by\_Gender'

'LeadConvRate\_by\_Credit\_Product'

'LeadConvRate\_by\_Channel\_Code'

'LeadConvRate\_by\_Avg\_acc\_bal\_X\_bins'

Similarly relative features were also added based on Target and the variables were transformed from train set to test set

**Statistical Features**

'Avg\_Acc\_bal\_by\_region\_code'[‘min’,’max’,’mean’,’std’]

Avg\_Acc\_bal\_by\_Age\_X\_bins [‘min’,’max’,’mean’,’std’]

'Avg\_Acc\_bal\_by\_Channel\_Code'[‘min’,’max’,’mean’,’std’]

'Avg\_Acc\_bal\_by\_Occupation’[‘min’,’max’,’mean’,’std’]

'Avg\_Acc\_bal\_by\_vintage'[‘min’,’max’,’mean’,’std’]

**Model Building:**

Tried models like Logistic regression and decision trees, The predictions were not that good, So used a LGBM model with stratified k fold and did hyperparameter tuning. Params used are

params = {

'n\_estimators':10000,

'learning\_rate': 0.1,

*#'is\_unbalance': 'true', # replaced with scale\_pos\_weight argument*

'num\_leaves': 7, *# 2^max\_depth - 1*

'max\_depth': 3, *# -1 means no limit*

'metrics':'auc',

'min\_child\_samples': 100, *# Minimum number of data need in a child(min\_data\_in\_leaf)*

'max\_bin': 100, *# Number of bucketed bin for feature values*

'subsample': 0.632, *# Subsample ratio of the training instance.*

'subsample\_freq': 1, *# frequence of subsample, <=0 means no enable*

'colsample\_bytree': 0.55, *# Subsample ratio of columns when constructing each tree.*

'min\_child\_weight': 0, *# Minimum sum of instance weight(hessian) needed in a child(leaf)*

'scale\_pos\_weight':3.215704776283283 *# because training data is extremely unbalanced*

}

Selected Top 50-60 features based on Feature importance and retrained the model. The model improved from 87.4349 to 87.4592 using LGBM feature selected variables.

Similar models were built using XGB and Catboost but LGBM gave better accuracy than other models and Voting classifier also did not improve the performance. Hence chosen LGBM classifier

Chart

Description automatically generated