Credit Card Lead Prediction

Objective: The Happy Customer Bank wants to cross sell its credit cards to its existing customers. The bank has identified a set of customers that are eligible for taking these credit cards.

Now, the bank is looking for help in identifying customers that could show higher intent towards a recommended credit card, given:

- Customer details (gender, age, region etc.)
- Details of his/her relationship with the bank (Channel_Code,Vintage, 'Avg_Asset_Value etc.)

So, our target variable is categorical variable having 2 values: 1 means customer is interested for credit card and 0 vice versa having event rate of 23.72% in training data.

Data Dictionary

Train Data

Variable	Type	Definition
ID	Unique ID	Unique Identifier for a row
Gender	Categorical (2)	Gender of the Customer
Age	Continuous	Age of the Customer (in Years)
Region_Code	Categorical (35)	Code of the Region for the customers
Occupation	Categorical (4)	Occupation Type for the customer
Channel_Code	Categorical (4)	Acquisition Channel Code for the Customer (Encoded)
Vintage	Continuous	Vintage for the Customer (In Months)
Credit_Product	Categorical (2+NA=3)	If the Customer has any active credit product (Home loan, Personal loan, Credit Card etc.)
Avg_Account_Balance	Continuous	Average Account Balance for the Customer in last 12 Months
Is_Active	Categorical (2)	If the Customer is Active in last 3 Months

Is_Lead(Target)	Target	If the Customer is interested for the Credit Card 0 : Customer is not interested 1 : Customer is interested

Data Pre-processing/Feature Engineering:

1) Null value Treatment: = We have null values in column Credit_Product both in traing test data, since it is categorical column having category “No” and “Yes”, we added another category of null values named “null_val”.
2) Then we did some EDA: we plotted histogram for continuous variables against our target variable, and what we found out that younger age customers are less interested in credit card compared to older people.
3) We also performed bar plot for categorical var, just to check frequency distribution of target variables across different categories in categorical variable.
4) Then we checked correlation between all the 3 continuous feature the highest correlation is 0.63 between Age and Vintage so we don’t need to drop any variable
5) Then we did one hot encoding for all the 6 categorical variable using panda get dummies we got 50 variable having value 1 and 0 only.
6) Then we checked correlation after this for all the 53 features no one correlation is more than 0.7 so we don’t need to drop any variable
7) Then we did outlier treatment for continuous variable:
   a) To check the outlier we plotted box plot and what we observed is that Avg_Account_Balance has lot of outliers
   b) To remove the outliers we applied flowing and capping based on below formula:
      1) Flooring: a=np.percentile(df3[i],25)-1.5*(np.percentile(df3[i],75)-np.percentile(df3[i],25))
      2) Capping: b=np.percentile(df3[i],75)+1.5*(np.percentile(df3[i],75)-np.percentile(df3[i],25))
8) Then we split our training dataset into 2 parts one is development data called as dev(70% training data) and Intime validation data called as itv (30% data) and we already have otv data given.

Model Building:

1) First we tried logistic regression model and simple tree based model, but result wasn’t encouraging.
2) Then we moved to bagging (Random Forest) and Boosting (XGBoost) Technique and XGBoost showed promising results.
3) Then we went one step ahead into XGBoost and to tune the hyperparameter tried multiple combination for e.g 1000 iteration
4) The best result which we got is for the following hyperparameter values:

```
'learning_rate' : 0.05,
'booster' : 'gbtree',
'objective' : 'binary:logistic',
```

'max_depth' : 7,
'seed' : 155,
'colsample_by_tree' : 0.8,
'subsample' : 0.8,
'scale_pos_weight' : 0.9,
 'gamma' : 0.01,
 ‘num_boost_round’:238

5) We got the AUC ROC for Training : 0.89 , OTV: 0.874 and ks = 62.74 at 3rd decile