**Approach**

The following are the main steps that have been taken for this challenge:

1. Data Loading & Cleaning:

Training Dataset have been checked for any null or repeat values and necessary steps have been taken to tackle the same.

1. Univariate Analysis:

All the predictors have been plotted against the response variable (i.e. *Is\_Lead*) and the important difference makers have been noted. New features have been created based on these difference makers.

1. Bivariate Analysis:

Combinations of following predictors and response variable have been checked for any specific slice of data, with high lead rate:

1. Gender vs occupation vs Is\_lead
2. Gender vs channel vs Is\_lead
3. Gender vs Vintage\_cat\_2 vs is\_lead
4. Gender vs Age\_cat\_1 vs is\_lead
5. Gender vs is\_active vs Is\_lead
6. Region\_Code\_1 vs Occupation vs Is\_lead
7. Avg\_Account\_Balance vs Age vs Is\_Lead

Based on the findings, features have been engineered to catch the sign of lead.

1. Frequency Encoding of Categorical Features:

FE of all the categorical features have been created and hence generated the new features.

1. Dummy Variable Creation of Categorical Features:

Dummy variables have been created for all the categorical features.

1. Train-Test Splitting:

Training dataset has been divided into two splits viz. X\_train and X\_test. The ratio of 85:15 has been maintained for the same. The reason for creation of such bifurcation is to keep track of the overfitting. Both the splits have been stratified so as to avoid any class imbalance.

1. Normalization of Continuous Features:

Post train-test splitting, normalization of all the continuous features have been done and hence made all of them in the range of 0-1. This is done to avoid over valuation (in terms of weight) of any specific feature, due to its large range.

1. Model Building:

Following models have been built and tested for the prepared dataset:

1. Logistic Regression
2. Neural Network
3. CatBoost
4. XGBoost
5. LightGBM
6. Random Forest

No hyperparameter tuning has been done and instead default hyperparameters have been used for all the above models. The following *roc\_auc\_score* has been obtained on the test split:

|  |  |
| --- | --- |
| **Model** | ***roc\_auc\_score*** |
| Logistic Regression | 0.86026 |
| Neural Network | 0.86688 |
| CatBoost | 0.87207 |
| XGBoost | 0.87125 |
| LightGBM | 0.87148 |
| Random Forest | 0.84945 |

1. Model Blending:

Output of each model is *Probability of Customer showing interest (class 1).* Various combinations of models have been tried. I have tried averaging the output of different models. The best results are obtained with the combination of *LightGBM* & *CatBoost* and hence the same has been adopted for building the final model.

***\* Please note that the above is only the overview of the submitted model. The detailed analysis and step by step approach with logical rationale have been brought in the submitted jupyter notebook.***