Analytics Vidhya

JOB-A-THON

Himanshu Rai

Agenda

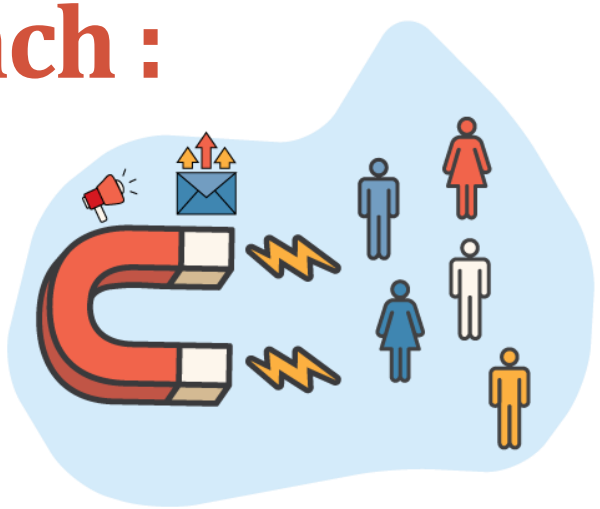
- Problem Statement
- Why?...Business Approach
- What's Provided?
- Approach towards the problem
  - Understand the data
  - Exploratory Data Analysis
  - Preprocess the data
  - Visualization of data
  - Cleaning data
  - Model preparation
- Different model metrics summary

Problem Statement

Credit Card Lead Prediction

- Happy Customer Bank is a mid-sized private bank that deals in all kinds of banking products, they 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
- **Objective to predict Is_Lead**: Bank is looking for your help in identifying customers that could show higher intent towards a recommended credit card.

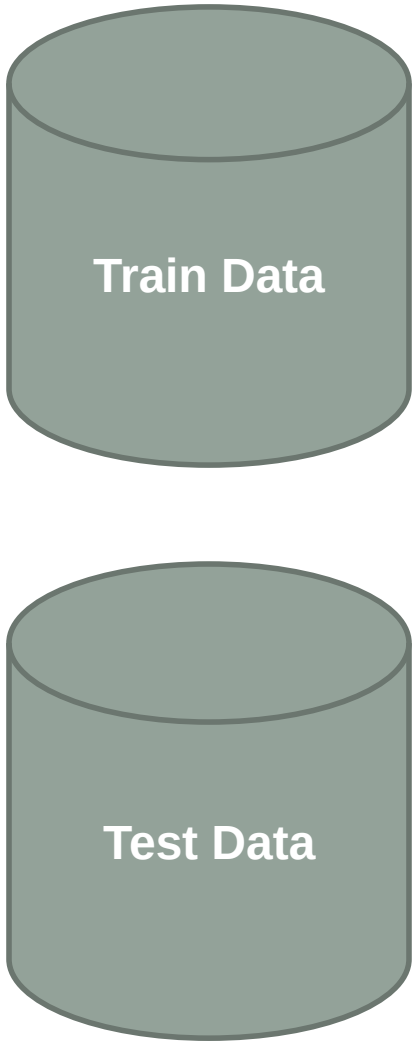
WHY? Business Approach :



- Effective targeting
- Customer care
- Personalized rewards
- New revenue opportunities
- Helps in prioritizing the target customers instead of searching.
- Helps in minimizing the costing of advertising as issuers can directly be in touch with most probable leads.
- Which customers are more likely to opt for Credit Cards when contacted.
- Better sense of the exponentially increasing data. (both transactional and behavioral)

What's Available?

- Train Data
- Test Data



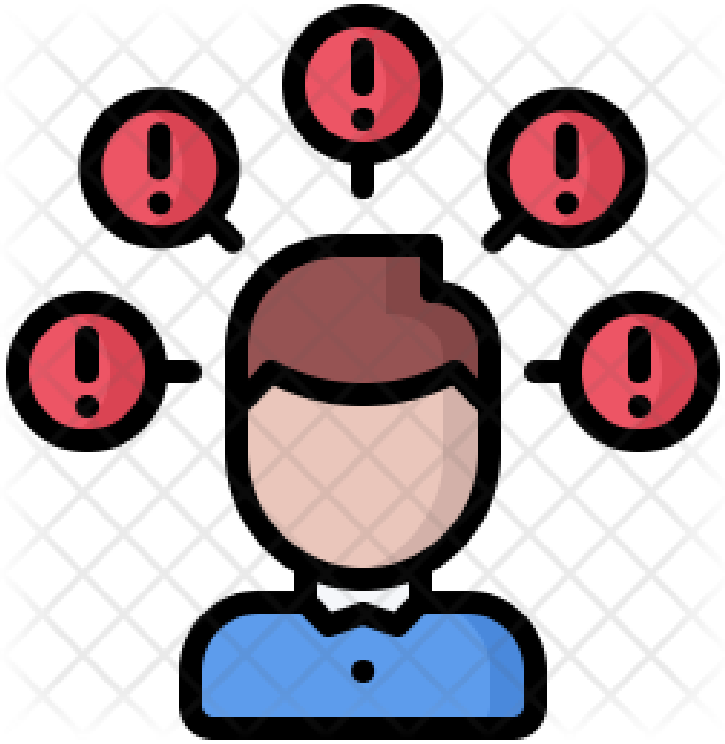
Train Data Dataset Description

We have 245725 rows and 11 columns in Train set

Variable	Description
ID	Unique Identifier for a row
Gender	Gender of the Customer
Age	Age of the Customer (in Years)
Region_Code	Code of the Region for the customers
Occupation	Occupation Type for the customer

Variable	Description
Channel_Code	Acquisition Channel Code for the Customer (Encoded)
Vintage	Vintage for the Customer (In Months)
Credit_Product	If the Customer has any active credit product (Home loan, Personal loan, Credit Card etc.)
Avg_Account_Balance	Average Account Balance for the Customer in last 12 Months
Is_Active	If the Customer is Active in last 3 Months
Is_Lead(Target)	If the Customer is interested for the Credit Card 0 : Customer is not interested 1 : Customer is interested

Approach Towards The Problem

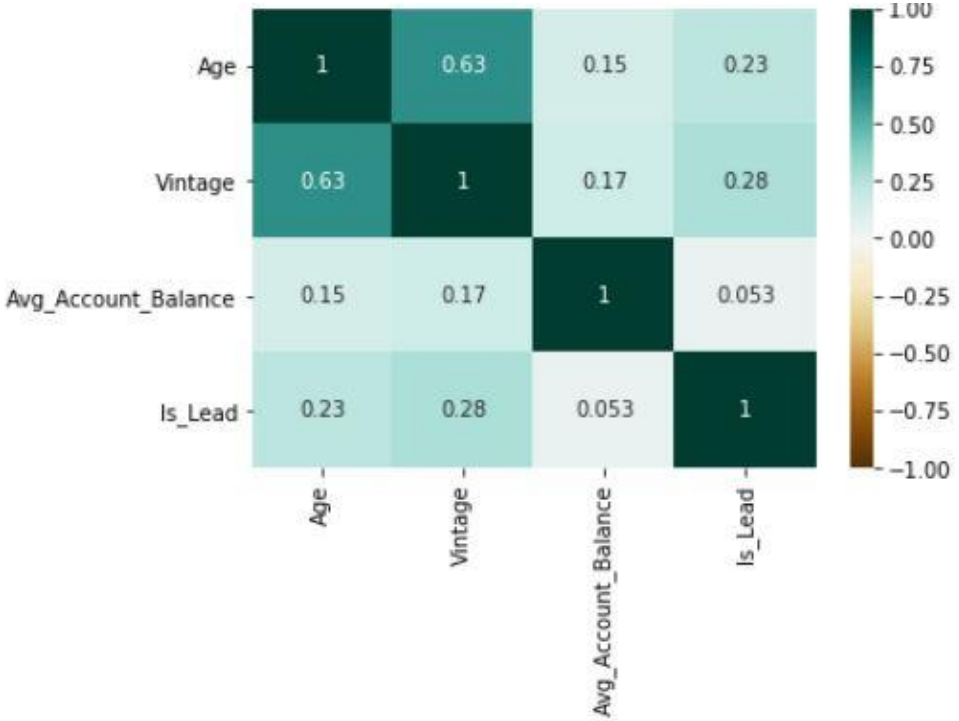


Flow of Approach

1. Exploratory Data Analysis
2. Preprocessing of data
3. Chi – Square Dependency Test
4. Visualization for Further Analysis
5. Data Cleaning
6. Splitting into Train and Validation
7. Preperation of Model building
8. Model Building

Exploratory Data Analysis (EDA)

- Used ***Dtale*** **library function** to analyze all variables.
- Checked **multicollinearity** with correlation heat map plot.



- Analyzed data for :
  - Shape
  - Statistical summary
  - Data types
  - Number of Unique values
  - Number of missing values
  - Target analysis
- Checked multicollinearity with correlation heat map plot.

Preprocessing Of Data

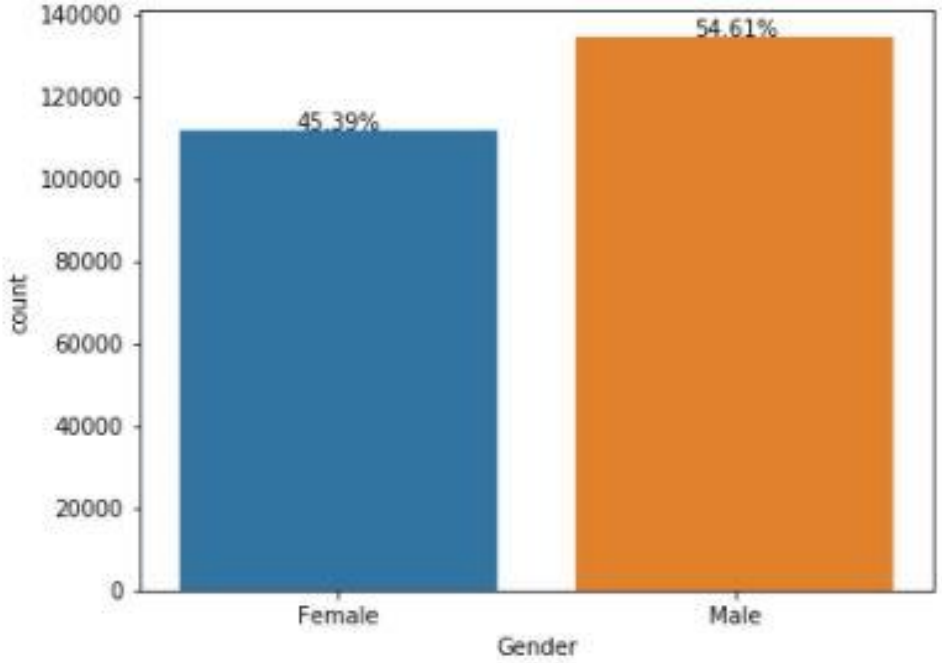
- Type casted datatypes to different variables as categorical and numerical based on unique values and domain knowledge.

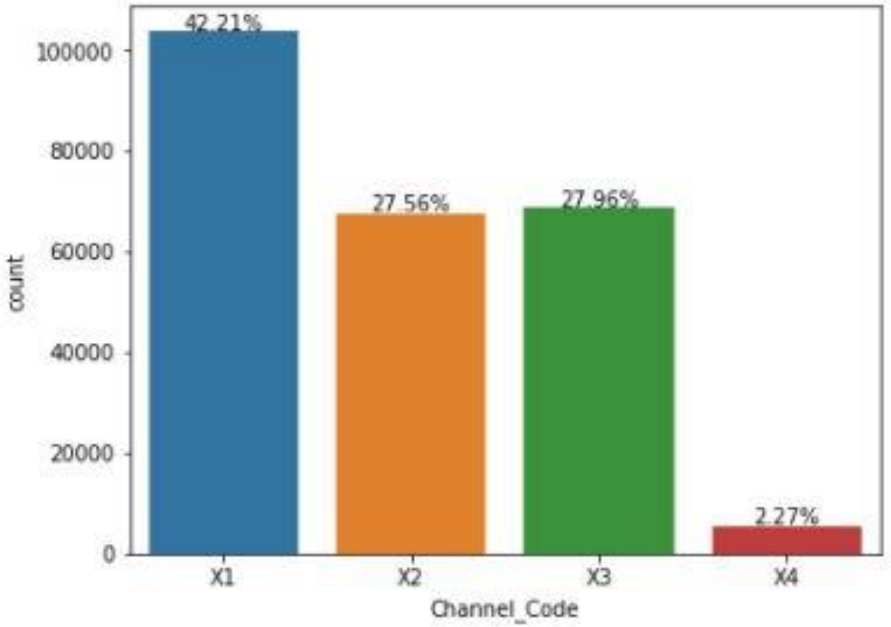
- Checked for unnecessary variables based on:
  - Zero variance
  - High Cardinality
  - All unique values
  - Duplicate Columns
  - Duplicate rows

- Chi-square Dependency Test : All variables are dependent.

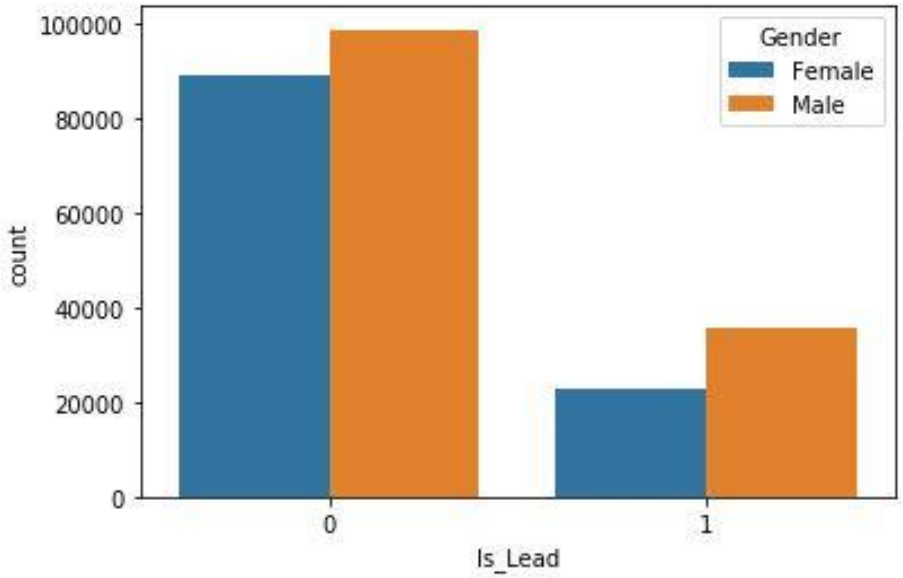
Visualization

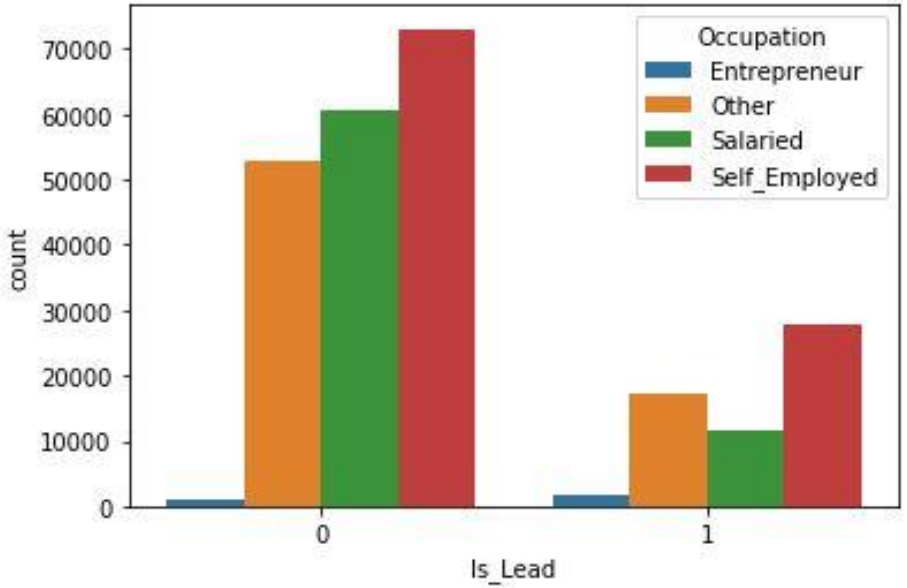
Univariate Analysis (some plots)



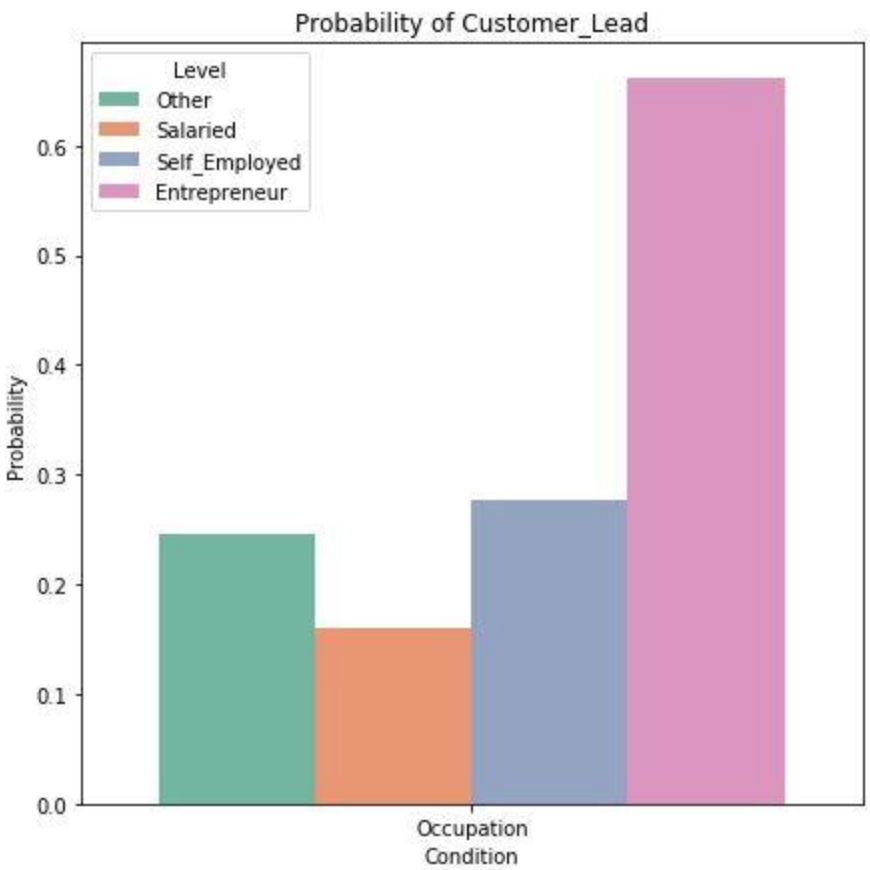


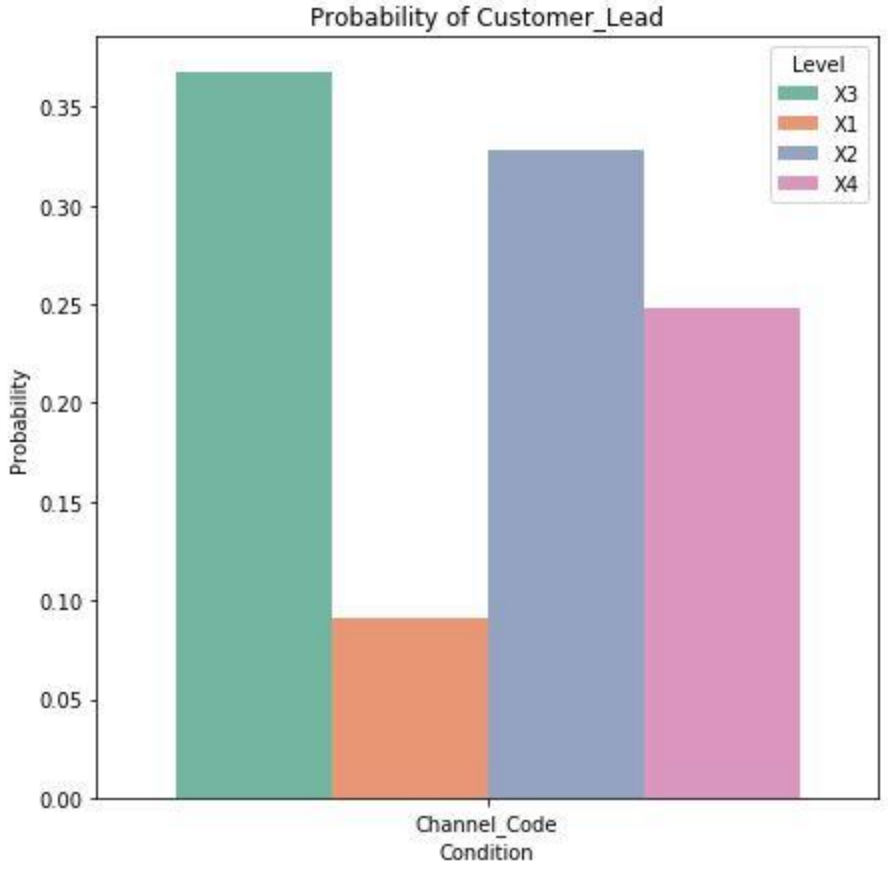
Bivariate Analysis (some plots)



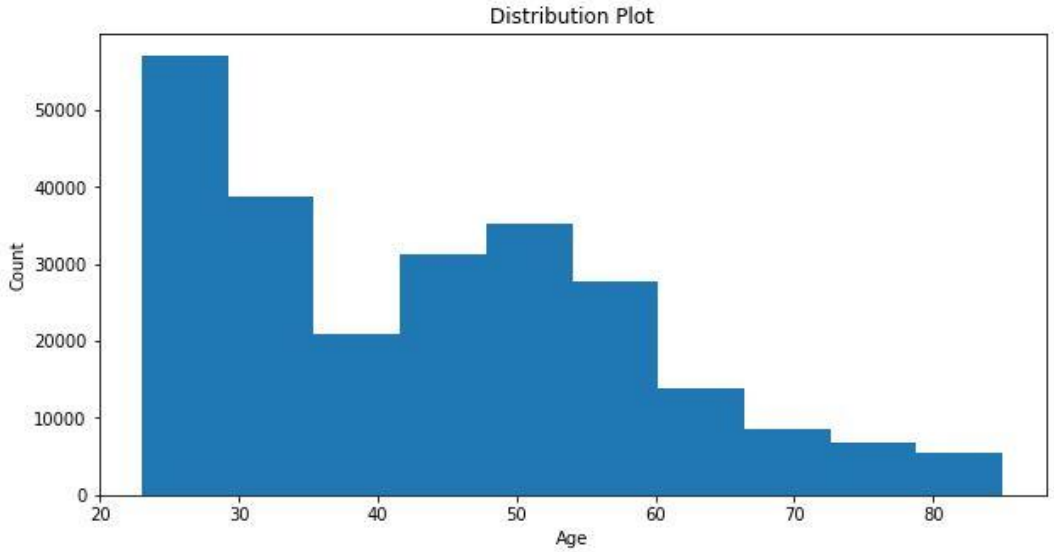


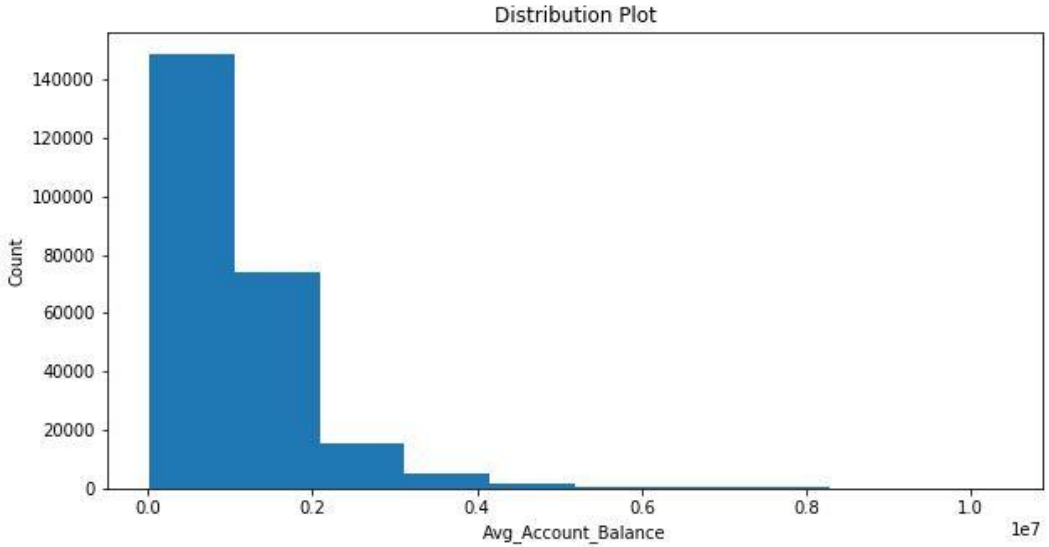
Probabilistic Analysis (some plots)



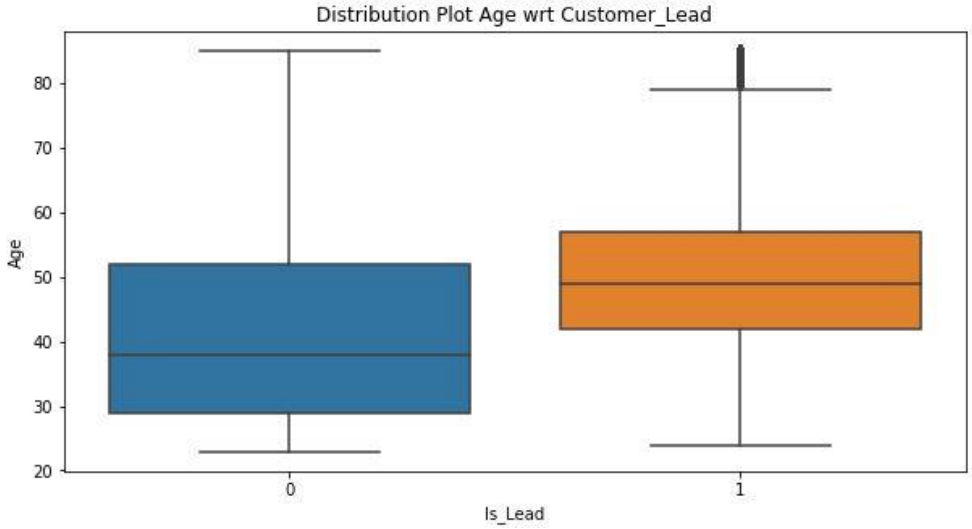


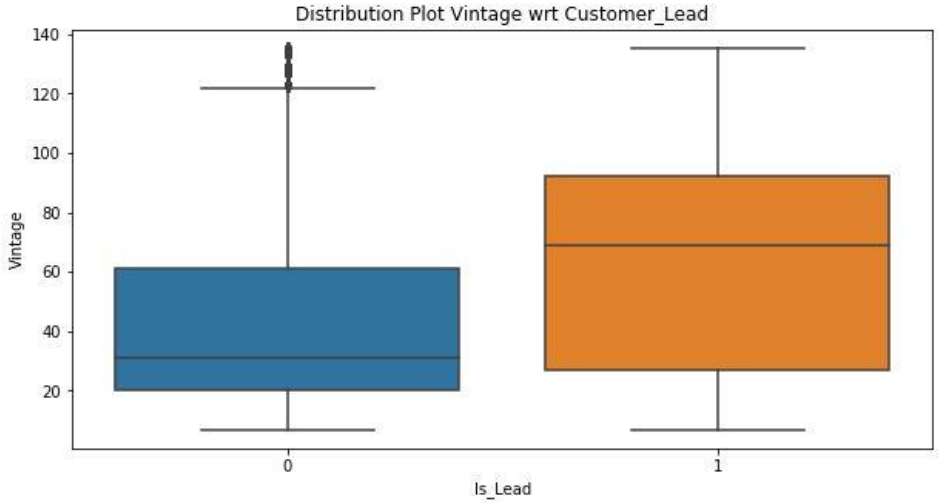
Histogram – Distribution(some plots)





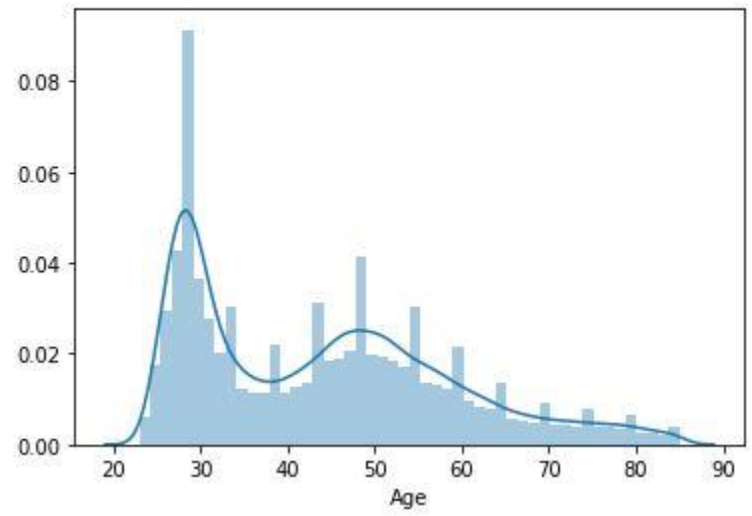
Boxplot (some plots)



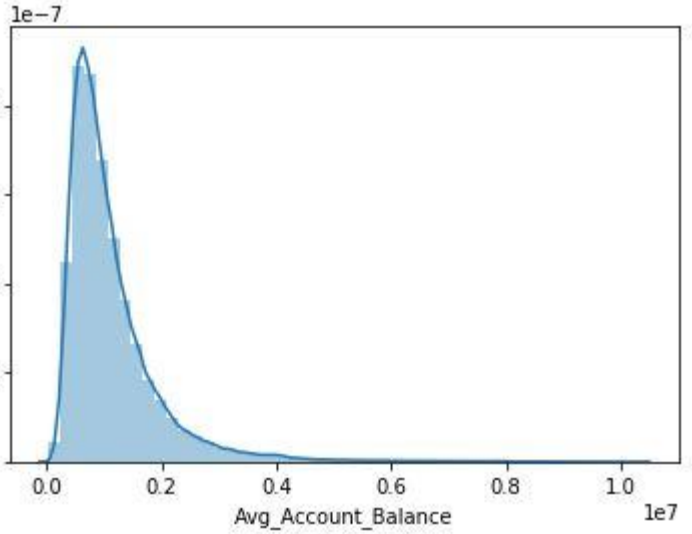


Distplots (some plots)

Age
0.6189884489476856



Avg_Account_Balance
2.9687083932770477



Analysis – Target Variable

- **Target variable : Is_Lead**

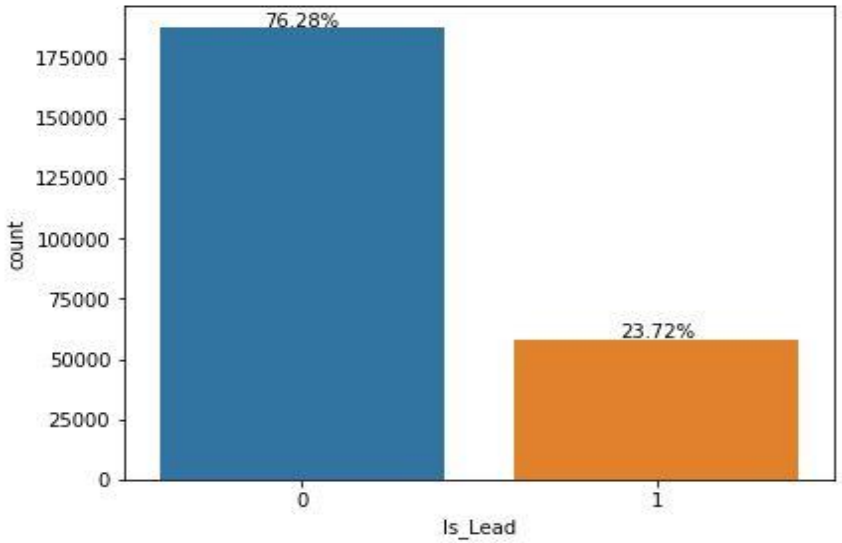
- 0 : If customer is not interested

- 1 : If customer is interested

Value Counts :

```
The percentage distribution target classes is as below:

0    0.762792
1    0.237208
Name: Is_Lead, dtype: float64
```



- Is_Lead has class- imbalance of ratio *76 : 24 (~approx.)*

Model Preparation

Model Preparation Steps :

1. Split data into train and validation (~0.25)
2. Removed **Outliers**
3. Checked and Imputed **Missing values** using SimpleImputer.
4. Checked **Multicollinearity** after standardization.
5. **Categorical Encoding**
   - Label Encoding (Gender, Is_Active and Credit_Product
   - Feature Encoding (Region_Code)
1. Initiated Pipeline
   - **PowerTransformer** (Handles Skewness and scaling)
   - **One Hot Encoding**

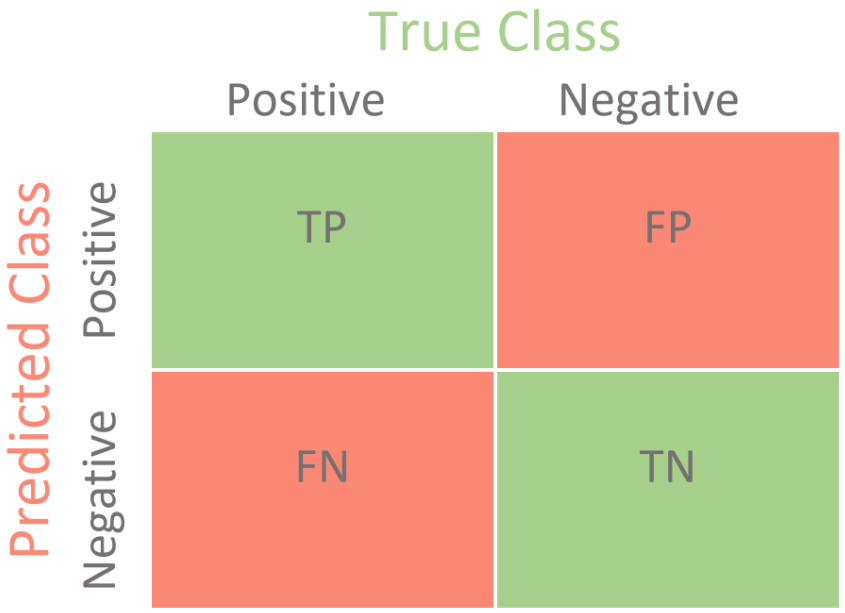
Model Building

Performance Metric

- **Performance Metric : Roc_Auc_Score**

Why Roc_Auc_Score ? :

Gives the measure of the ability of a classifier to distinguish between Classes.



Model Results

Model	Train_Accuracy	Train_Recall	Train_Precision	Train_F1_Score	Test_Accuracy	Test_Recall	Test_Precision	Test_F1_Score	Train_roc
LogisticRegression	0.780827	0.103784	0.789043	0.183439	0.781433	0.105682	0.795866	0.186588	0.547577
LogisticRegression_Balanced	0.617707	0.757068	0.356139	0.484405	0.615884	0.756313	0.354749	0.482964	0.665718
Lasso	0.616714	0.757480	0.355495	0.483893	0.614794	0.755833	0.353920	0.482098	0.665209
DecisionTree	0.583223	0.837840	0.344410	0.488155	0.578819	0.835781	0.341531	0.484910	0.670942
DecisionTree_BestParameters	0.581194	0.863071	0.346379	0.494356	0.577224	0.861515	0.343870	0.491543	0.678304
Random Forest	0.698703	0.750435	0.423724	0.541626	0.695354	0.746775	0.420041	0.537662	0.716525
Random Forest imp	0.710646	0.731105	0.434654	0.545186	0.707677	0.728932	0.431263	0.541911	0.717694
DecisionTree	0.703407	0.735840	0.427311	0.540657	0.701117	0.733256	0.424699	0.537867	0.714581
Random Forest imp	0.669836	0.765372	0.398089	0.523758	0.668316	0.766607	0.396895	0.523011	0.702749
Random Forest imp	0.670736	0.779852	0.400381	0.529112	0.669277	0.780469	0.399179	0.528203	0.708328
Random Forest imp	0.711232	0.730762	0.435267	0.545573	0.708002	0.728315	0.431563	0.541977	0.717960
RF_CV	0.711639	0.731288	0.435753	0.546101	0.708165	0.728040	0.431716	0.542022	0.718408

Conclusion :

- Extracted Important features from LASSO and ExtraTreeClassifier.
- Build many models with balanced class weight and Grid search hyper parameter tuning.
- Many models seem to have high bias and some are highly over fitted.

- Most Stable Models are:
  - Decision tree with best parameters and ccp pruning(~71%)
  - Random Forest with GridSearch CV(~78%)
  - LGBM (~85%)

Himanshu Rai