BANK CHURN PREDICTION

BUSINESS PROBLEM STATEMENT

- Bank churn, or the loss of customers to other financial institutions, is a significant problem for banks because it can lead to a decline in revenue and profitability.Building a prediction model can help identify at-risk customers and prevent churn by taking targeted interventions to improve the customer experience.
- Ultimately, building a prediction model to prevent churn can help improve customer loyalty, increase revenue, and reduce the financial impact of customer loss.

PROJECT OBJECTIVES

1. What is the best model to predict and prevent Churn?
2. What are the most influential features that impact on the churn?
3. What metrics are the most suitable ones to evaluate model?
4. What is the financial benefit of using a model to prevent churn?
5. Deploying the model using Python Flask for real-time prediction

DATA DESCRIPTION

1. The credit card customer data in this project contains information on approximately 10,000 individuals, including demographic data such as age, education level, and marital status, as well as details about their credit card usage. There are a total of 23 columns and 10,127 rows in the dataset.

2. One notable aspect of this dataset is that it is slightly imbalanced, with only 16% of customers having cancelled their credit cards. This can make it challenging to train a model to accurately predict customer churn, as the model may be biased towards the majority class of customers who have not cancelled their credit cards. As a result, care should be taken when using this dataset to train machine learning models.

PROJECT STRUCTURE

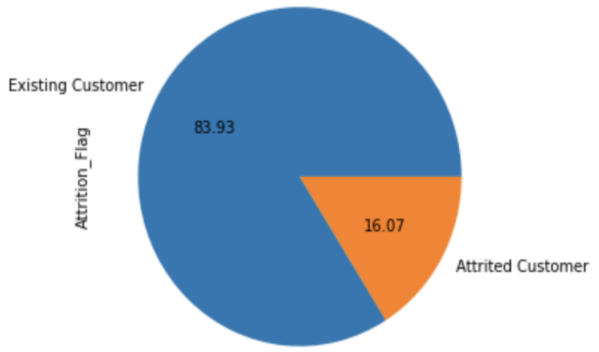
1. Exploratory Data Analysis & Feature Engineering
2. Model Building
3. Model Performance Evaluation
4. Model Improvement
5. Model Interpretation
6. Financial Benefits Evaluation
7. Model deployment

EXPLORATORY DATA ANALYSIS

```
target = df["Attrition_Flag"] # Target variable
```

```
target.value_counts()
```

```
Existing Customer    8500
Attrited Customer    1627
```



- The target variable in this dataset is "Attrition_Flag," which indicates whether a customer has left (1) or stayed with the company (0).
- In addition, this dataset is imbalanced, with the churning rate of 16%. This can make it difficult for the model to accurately learn and predict the patterns in the data, as the minority class is underrepresented.

EXPLORATORY DATA ANALYSIS

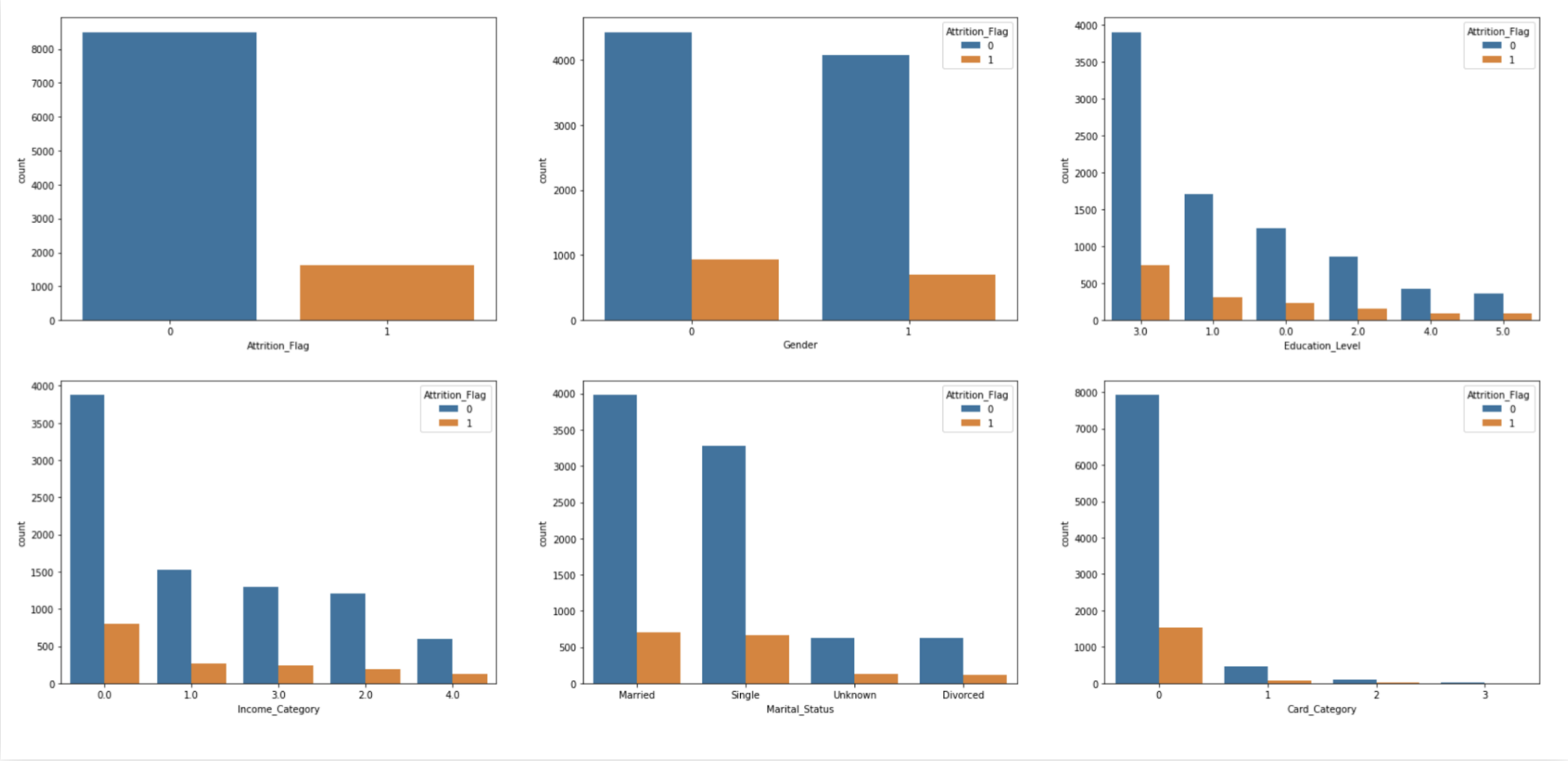
Null Value Detection

```
# Create a Boolean mask of null values
null_mask = df.isnull()

# Sum the mask to find the total number of null values
num_nulls = null_mask.sum()

print(num_nulls)
```

```
CLIENTNUM                                                                                                    0
Attrition_Flag                                                                                               0
Customer_Age                                                                                                 0
Gender                                                                                                       0
Dependent_count                                                                                              0
Education_Level                                                                                              0
Marital_Status                                                                                               0
Income_Category                                                                                              0
Card_Category                                                                                                0
Months_on_book                                                                                               0
Total_Relationship_Count                                                                                     0
Months_Inactive_12_mon                                                                                       0
Contacts_Count_12_mon                                                                                        0
Credit_Limit                                                                                                 0
Total_Revolving_Bal                                                                                          0
Avg_Open_To_Buy                                                                                              0
Total_Amt_Chng_Q4_Q1                                                                                         0
Total_Trans_Amt                                                                                              0
Total_Trans_Ct                                                                                               0
Total_Ct_Chng_Q4_Q1                                                                                          0
Avg_Utilization_Ratio                                                                                        0
Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_1    0
Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_2    0
dtype: int64
```

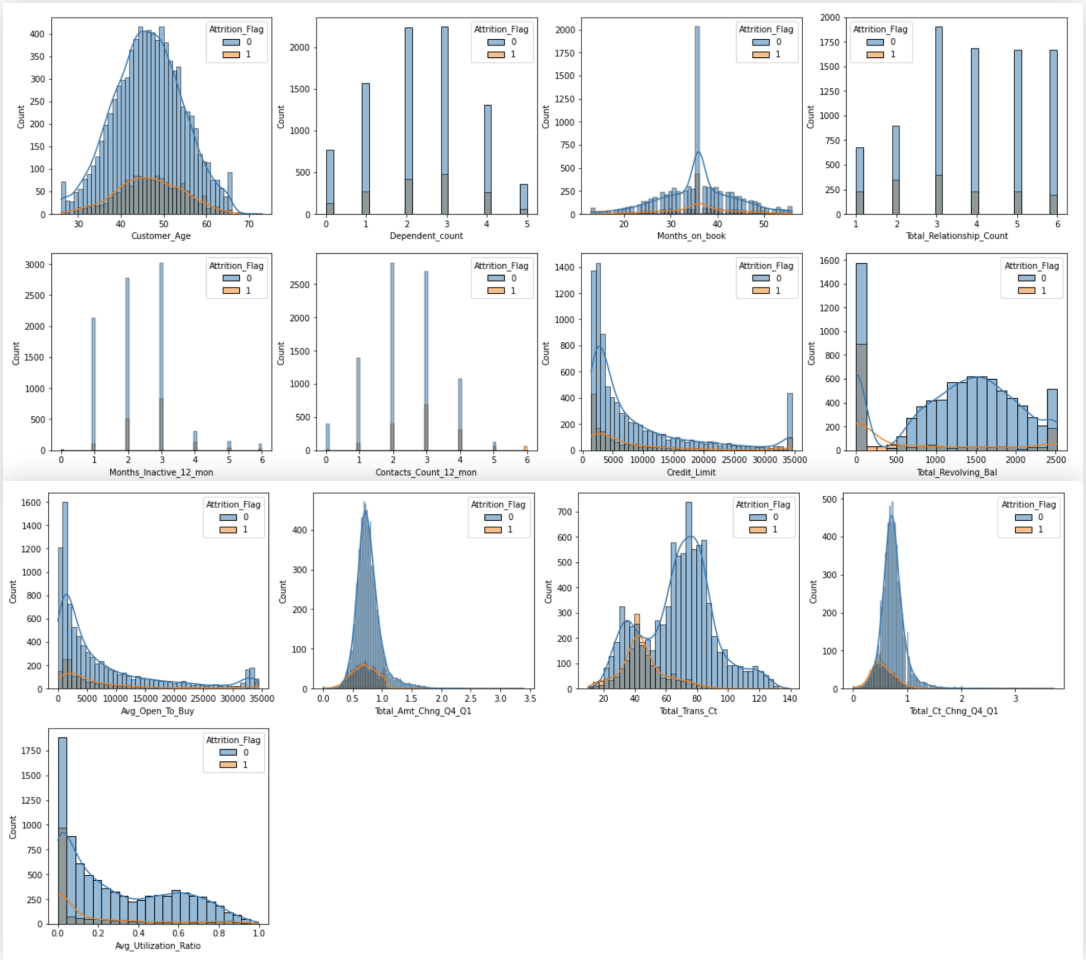
EXPLORATORY DATA ANALYSIS

Categorical Data



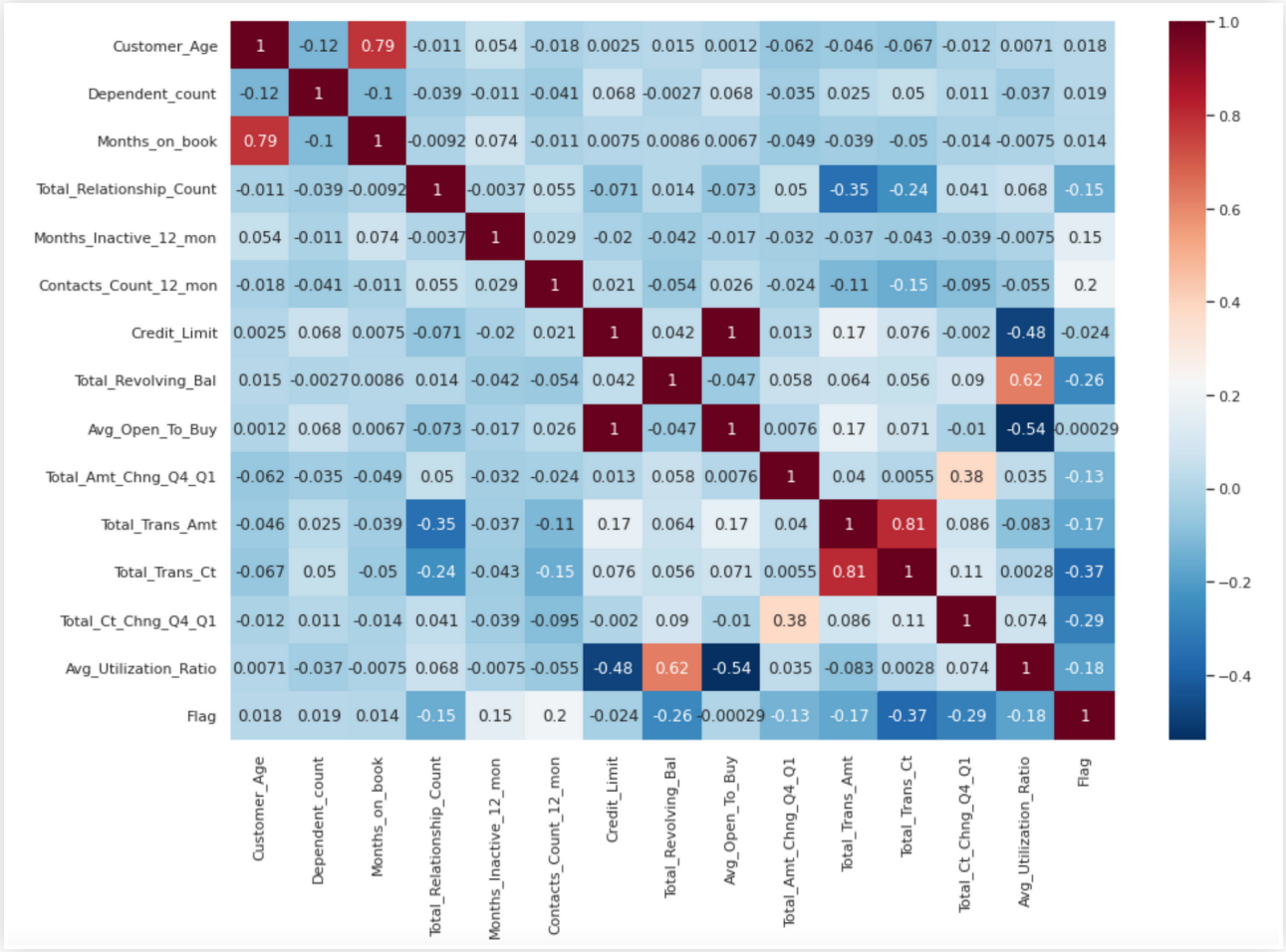
EXPLORATORY DATA ANALYSIS

Numerical Data



EXPLORATORY DATA ANALYSIS

Numerical Data

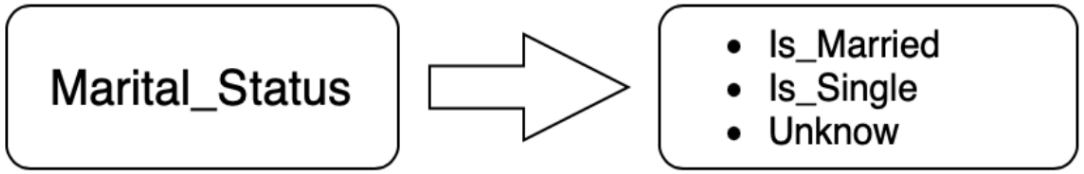


- The top 5 numerical features that correlated with target are:
  - Total_Trans_Ct
  - Total_Ct_Chng_Q4_Q1
  - Total_Revolving_Bal
  - Contacts_Count_12_mon
  - Avg_Utilization_Ratio

FEATURE ENGINEERING

Feature Creation & One-hot encode

```
df["Recolving_Bal_Per_Relationship"]=df["Total_Revolving_Bal"]/df["Total_Relationship_Count"]
```



EDA&FEATURE ENGINEER TAKEAWAYS:

1. This dataset is imbalanced, with a majority of observations belonging to label 0 and a minority belonging to label 1.

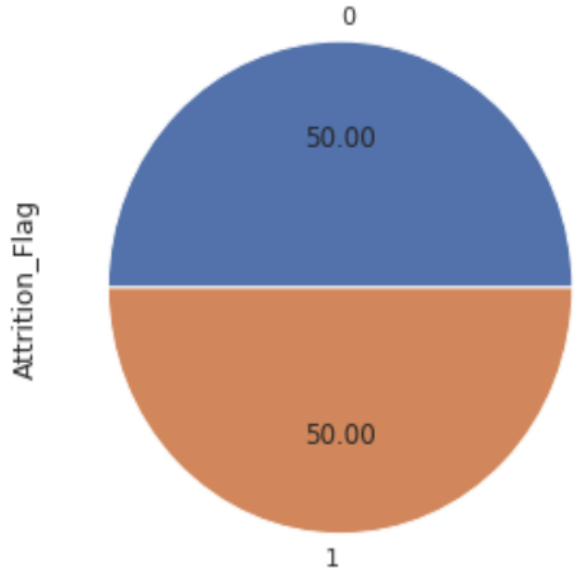
2. Missing values or major outliers are not being detected

3. Create new features & One hot encode

4. The top 5 features that correlated with target are:
   - Total_Trans_Ct
   - Total_Ct_Chng_Q4_Q1
   - Total_Revolving_Bal
   - Contacts_Count_12_mon
   - Avg_Utilization_Ratio

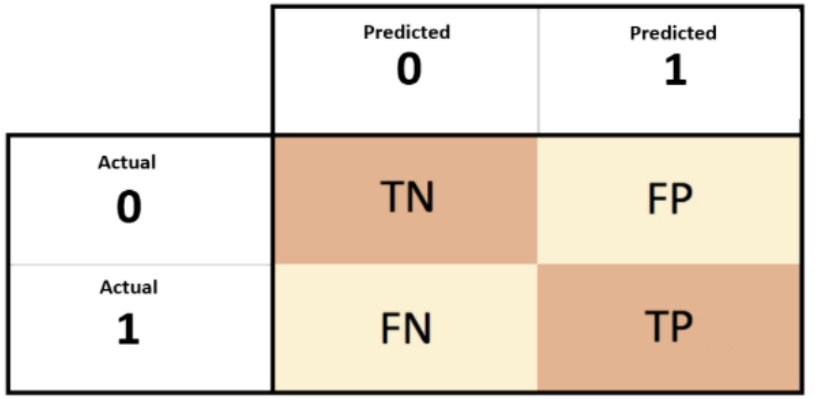
DATA OVER SAMPLING

Random over-sampling is a technique that is used to balance an imbalanced dataset by generating new synthetic samples from the minority class



EVALUATION METRICS

In this problem, the goal is to minimize the customer who actually left bank but the model fails to detect(FN). This is because a failure to detect a customer who has actually left (FN) can result in the bank losing money, while a false alarm (FP) does not have the same issue. Therefore, we will prioritize recall over precision.



MODEL PERFORMANCE EVALUATION

Model trained with imbalanced Data

Model	Recall	F1	PR AUC	ROC AUC
Logistic Regression	0.50	0.59	0.65	0.88
Random Forest	0.57	0.69	0.85	0.96
XG Boost	0.70	0.79	0.90	0.97

Model trained with balanced Data

Model	Recall	F1	PR AUC	ROC AUC
Logistic Regression	0.81	0.60	0.69	0.90
Random Forest	0.89	0.77	0.89	0.97
XG Boost	0.92	0.79	0.90	0.98

MODEL IMPROVEMENT

Hyper parameter Tuning

```
param_grid = {'max_depth': [3, 4, 5],
              'learning_rate': [0.01, 0.02, 0.03],
              'n_estimators': [100, 130, 150]}


from sklearn.model_selection import GridSearchCV

gscv = GridSearchCV(XGB, param_grid, cv=3)
gscv.fit(X_train_os, y_train_os)

# Print the best hyperparameters
print(gscv.best_params_)
```

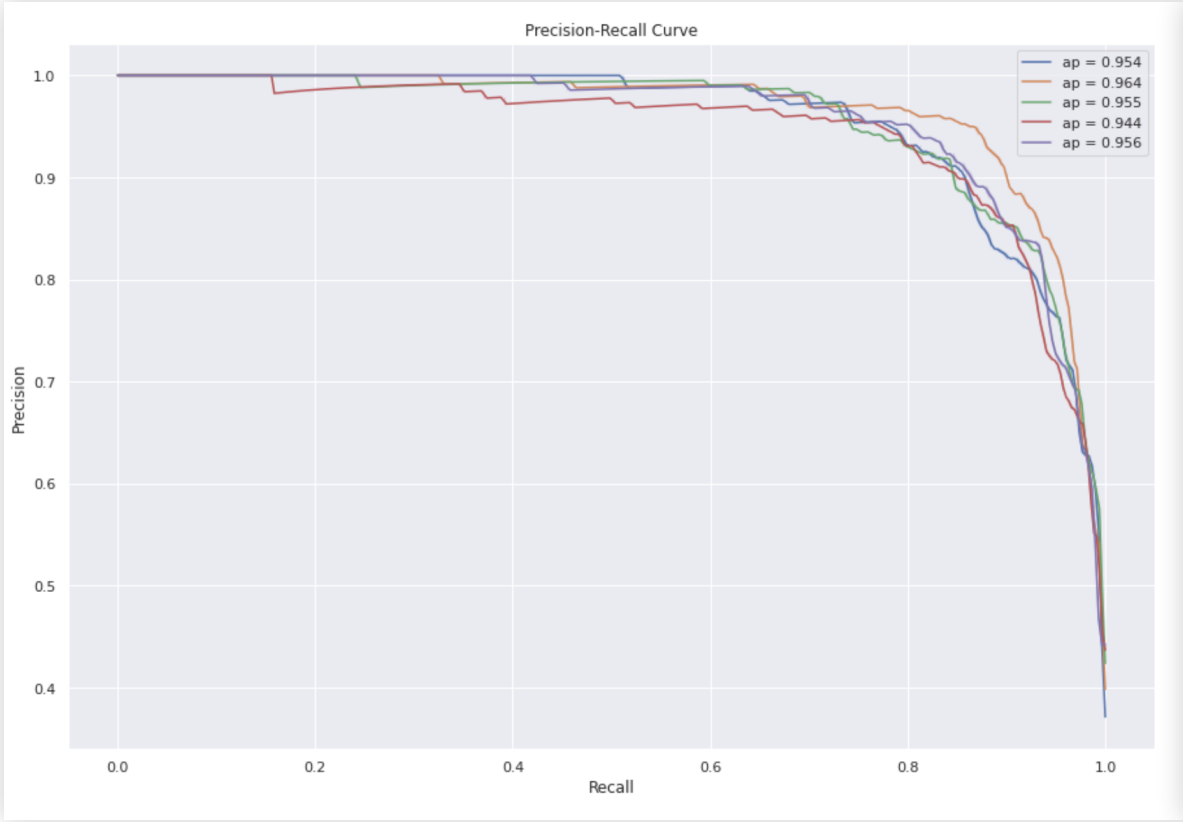
```
{'learning_rate': 0.03, 'max_depth': 5, 'n_estimators': 150}
```

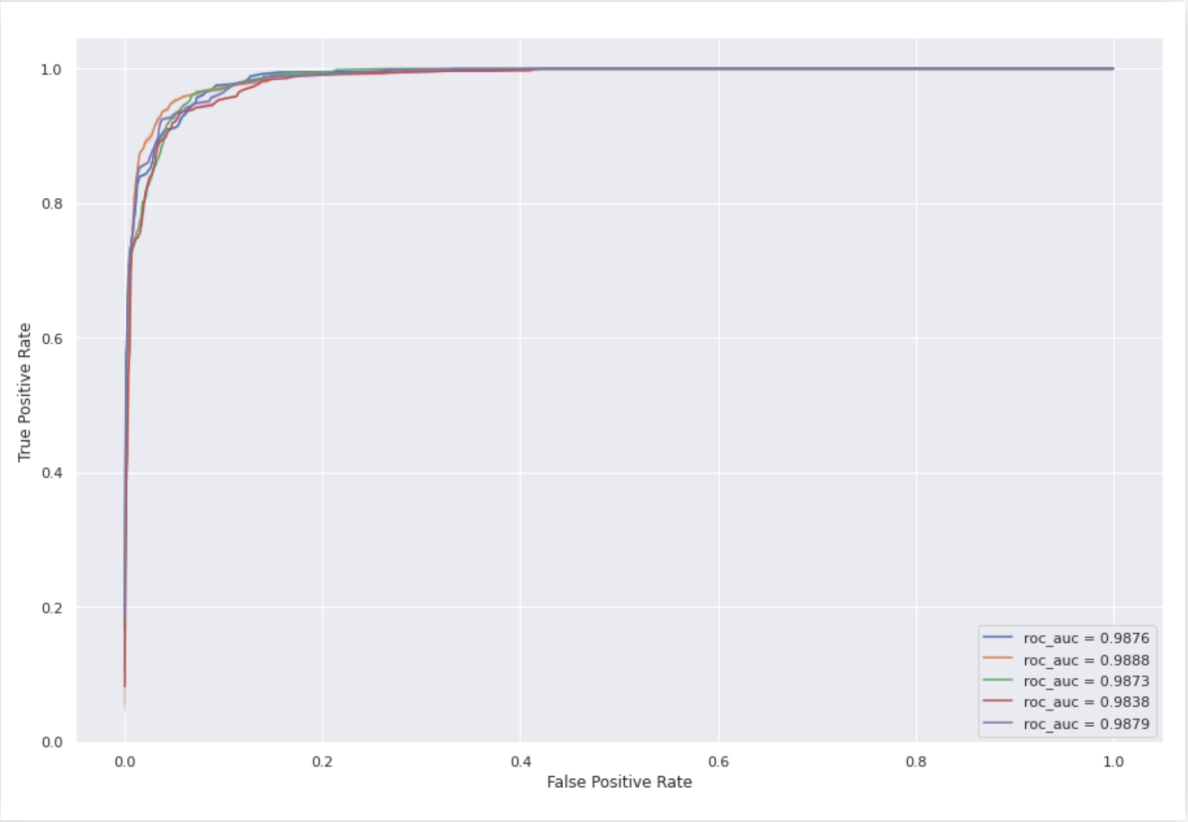
Retrain the model using best parameters

Model	Recall	F1	PR AUC	ROC AUC
XG Boost	0.94	0.87	0.95	0.99

MODEL IMPROVEMENT

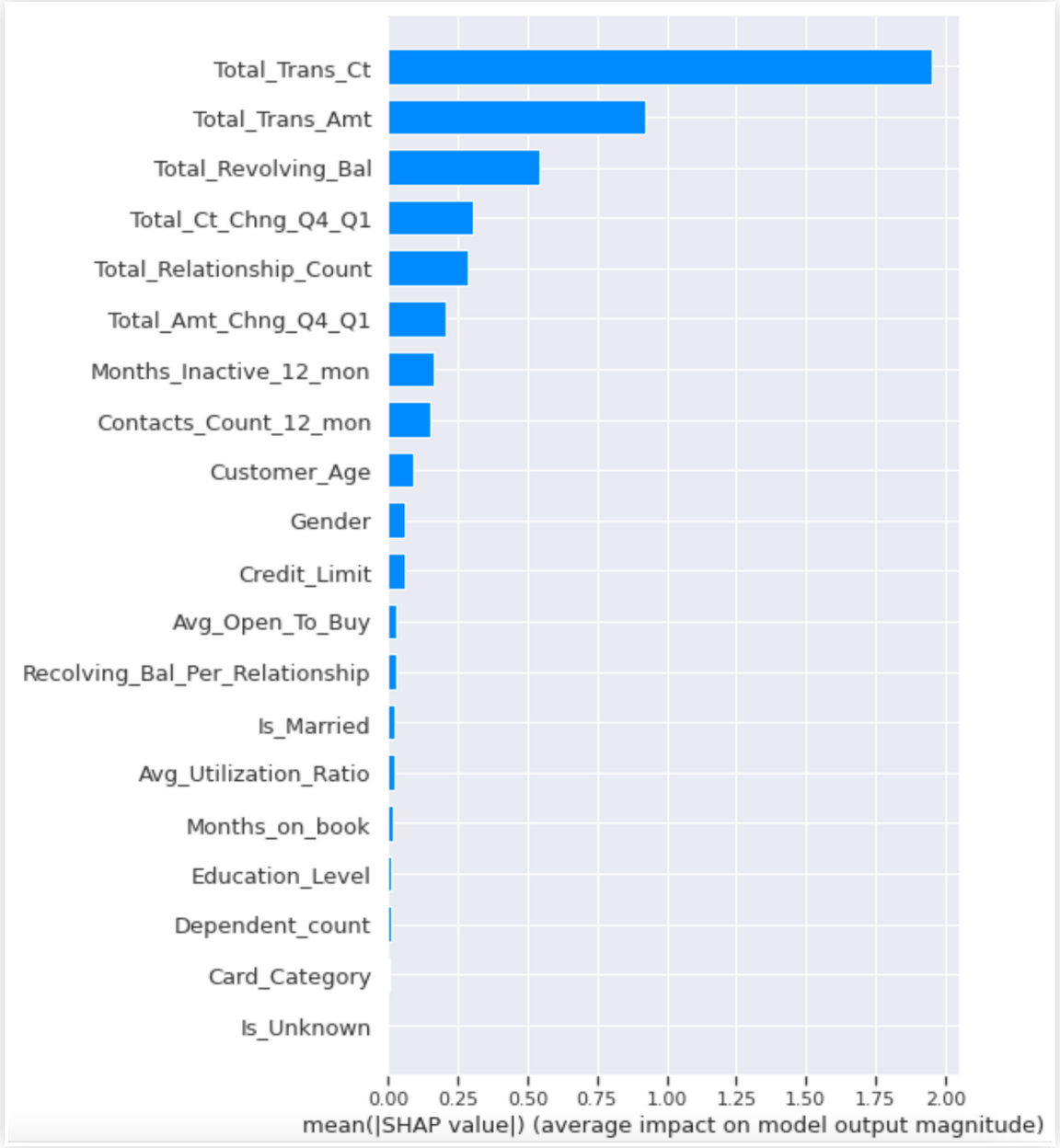
Cross Validation PR AUC VS. ROC AUC





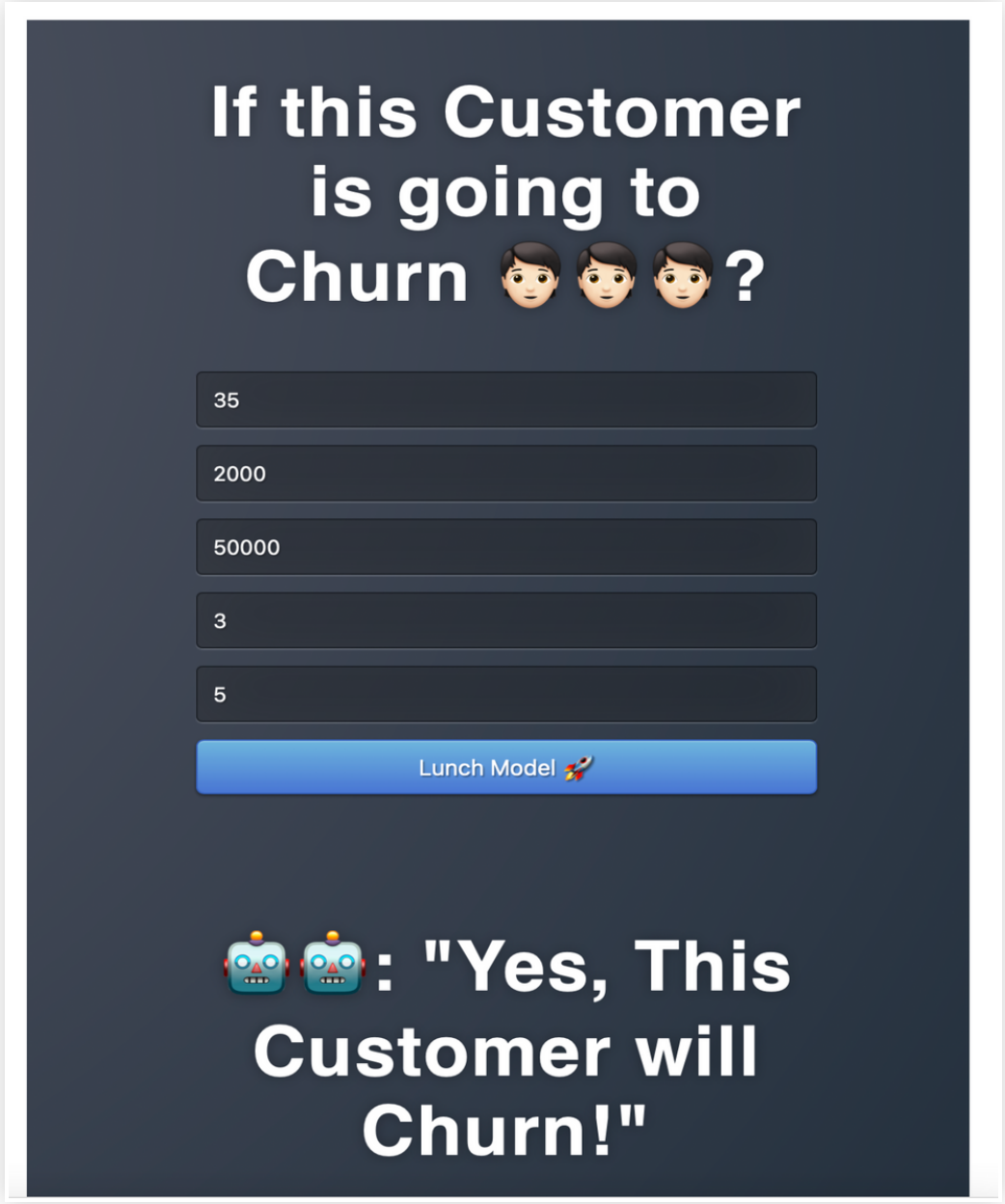
FEATURE IMPORTANCE

In this section, we used SHAP values to identify the most influential features in the model. We selected the top 5 features based on their SHAP values and included them in the model. This allowed us to evaluate the importance of each feature and determine which ones had the greatest impact on the model's performance.



DEPLOYMENT

In this part of the process, we will use the top 5 most influential features identified by SHAP values to build a XG-Boost model. We will then deploy this model using Python Flask to allow for real-time prediction for the new customers. This will enable us to quickly and efficiently make predictions using the model in a live setting.



CONCLUSION

The SHAP summary plot visualizes the importance of each feature in a model for predicting a specific outcome, with the x-axis representing the SHAP value and the y-axis ranking the features by importance.