Customer Churn Analysis

Zhirui Li
Data Science Initiative @ Brown
Dec 04, 2023
https://github.com/ZhiruiLi1/Customer_Churn_Analysis/tree/main

Outline

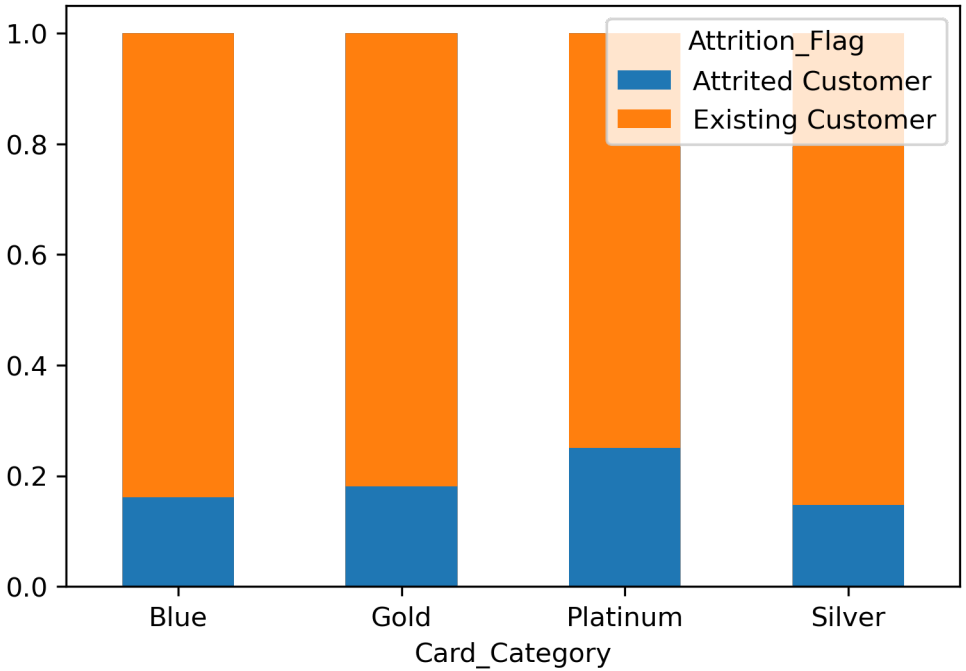
- Background, EDA, Preprocessing
- Cross Validation Pipeline
- Results
- Outlook

Background

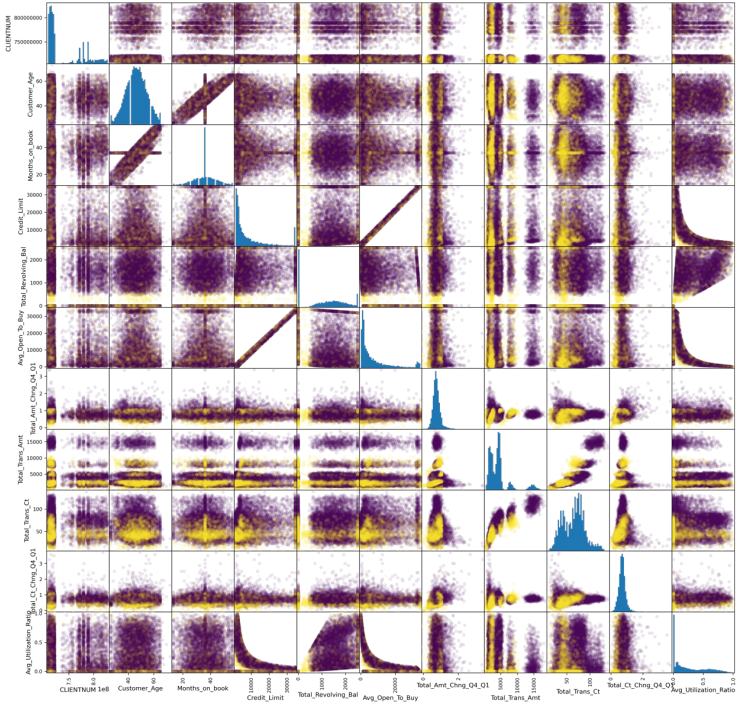
- **Customer Churn Prediction Analysis:** Addressing the challenge bank managers face as increasing numbers of customers leave their credit card services. The goal is to utilize available data to identify potential "churned customers" early, enabling proactive engagement and improved services to reverse their decision to leave potentially.
- **Importance and Data Overview:** Understanding and predicting customer churn is crucial, particularly without clear feedback. This project leverages a dataset from LEAPS with 21 columns and 10,127 data points, focusing on the 'Attrition_Flag' target variable. Effective churn prediction can empower customer experience teams, potentially reducing churn by 11% through timely intervention.

EDA

Categorical Variable: Card_Category

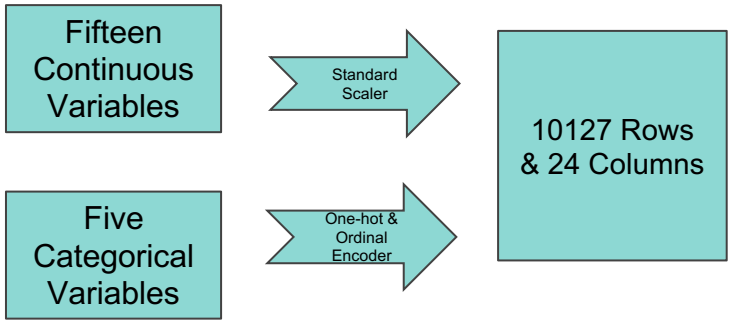


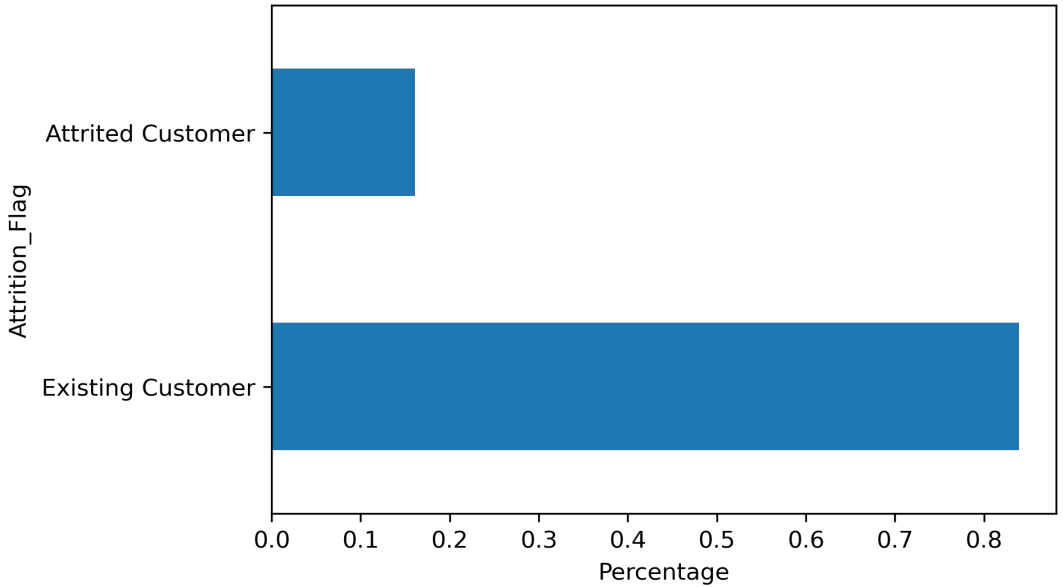
Continuous Variables



Preprocessing

- Imbalanced Dataset
  - Stratified Split
  - F2-beta Score





Cross Validation Pipeline

- **Random State Selection**: Choose five distinct random states to ensure reproducibility and robustness in the model's performance.
- **Data Splitting**: Employ stratified splitting to split the dataset into training (60%), validation (20%), and test (20%) sets, ensuring a proportional representation of class 1 samples across each subset.
- **Preprocessing**: Apply standard scaling to normalize numerical features and use one-hot & ordinal encoding to convert categorical variables into a format suitable for model training.
- **Hyperparameter Tuning**: Conduct an exhaustive grid search across various hyperparameter combinations to find the configuration that yields the best performance on the validation set.
- **Model Evaluation**: Finalize the optimal model based on grid search results and evaluate its effectiveness on the previously unseen test set to assess its generalization capability.

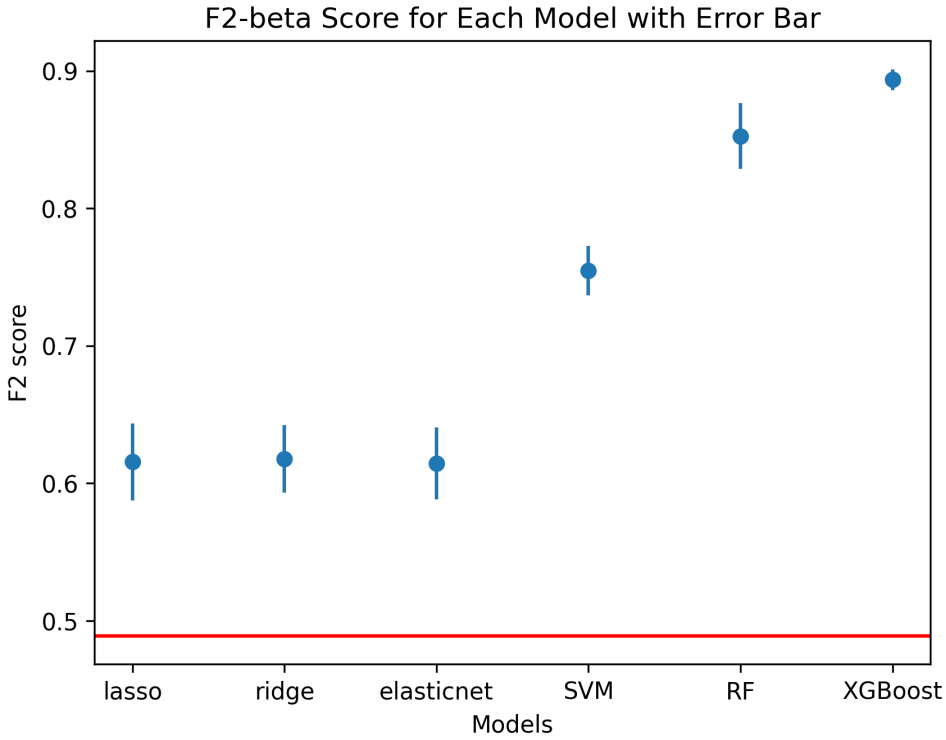
Parameter Tuning

- Models Chosen:
  - Lasso Logistic Regression
  - Ridge Logistic Regression
  - ElasticNet Logistic Regression
  - Random Forest
  - Support Vector Classifier
  - Extreme Gradient Boosting

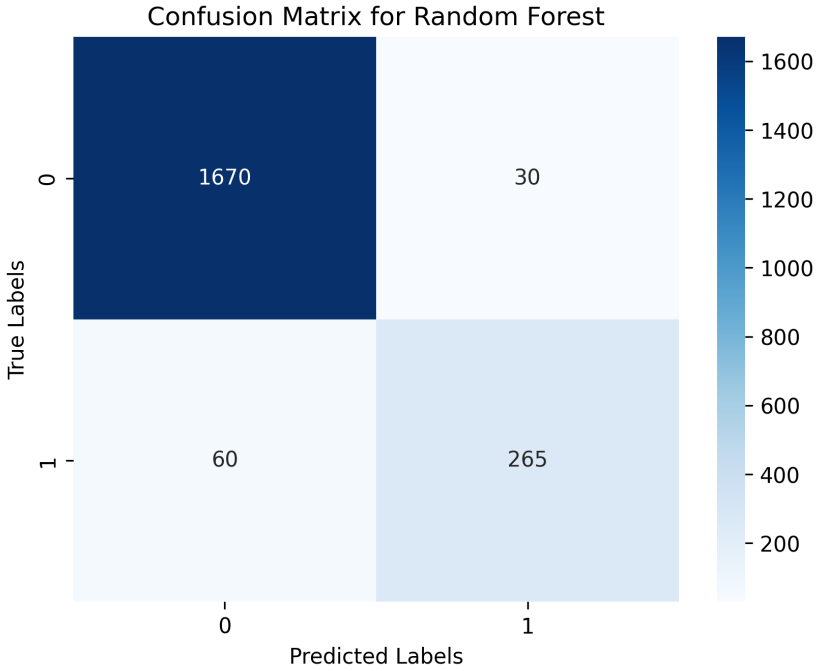
Model	Hyperparameter	Search Space
L1	C	10 values of logspace from -5 to 5 with base 10
L2	C	10 values of logspace from -5 to 5 with base 10
ElasticNet	C	10 values of logspace from -5 to 5 with base 10
	l1_ratio	[0.1, 0.3, 0.5, 0.7, 0.9]
RandomForest	max_depth	[1, 2, 3, 5, 10, 15, 20, 30, 50]
	max_features	[2, 5, 10, 15, 20]
SVM	C	[0.001, 0.01, 0.1, 1, 10, 100, 1000]
	gamma	[0.001, 0.01, 0.1, 1, 10, 100, 1000]
XGBoost	learning_rate	[0.03]
	n_estimators	[10000]
	min_child_weight	[1, 3, 5, 7]
	gamma	[0, 0.1, 0.2, 0.3, 0.4]
	max_depth	[2]
	colsample_bytree	[0.3, 0.4, 0.5, 0.7, 1]
	subsample	[0.5, 0.66, 0.75, 1]

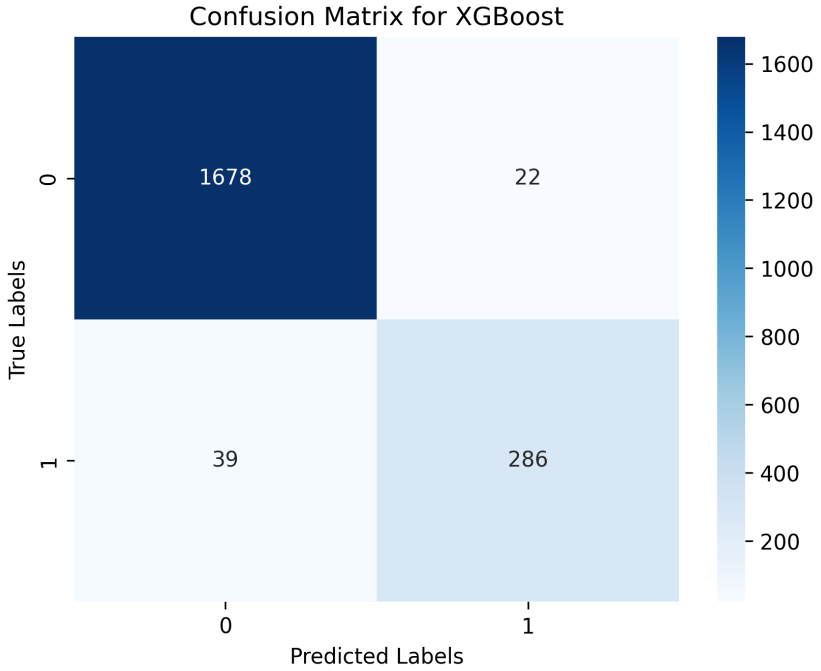
Results

	Model	F2 Beta Score (Mean ± Std)
0	Lasso Logistic Regression	0.616 ± 0.031
1	Ridge Logistic Regression	0.618 ± 0.027
2	SVM	0.755 ± 0.02
3	Random Forest	0.853 ± 0.027
4	XGBoost	0.894 ± 0.008
5	Baseline	0.489



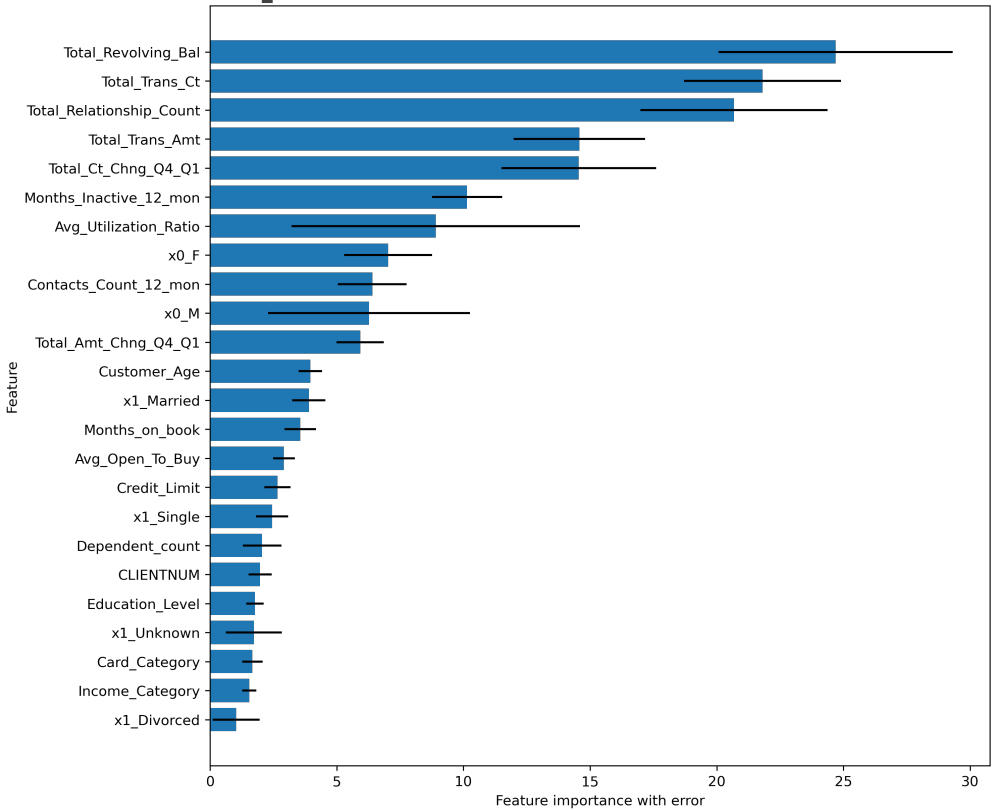
Results Continue



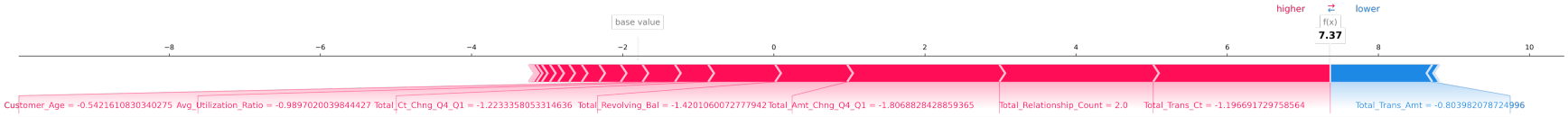


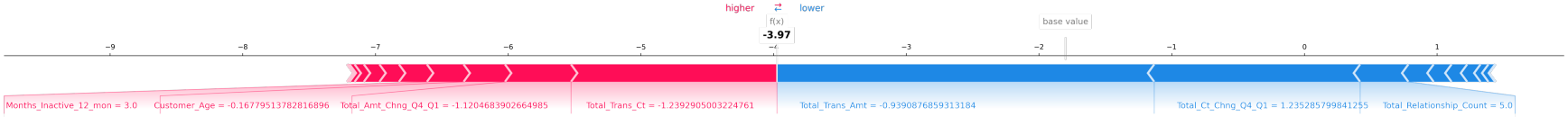
Global Feature Importance

- Total_Revolving_Bal is the most influential feature when predicting whether a customer will churn using XGBoost and Gain as the importance metric.



Local Importance





Outlook

- **Expand Dataset:** Increase dataset size to enhance model accuracy and generalization.
- **Enhanced Hyperparameter Tuning:** Explore a broader range of hyperparameters for optimal model performance.
- **Advanced Models:** Experiment with complex models such as neural networks for potentially better predictions.
- **Interpretability:** Conduct more literature reviews or consult domain experts for feature engineering.

Q & A

THANK YOU!