Predicting Churning Customers

Final Project Report for Data 1030, Fall 2021 at Brown University

Supervised by Prof. Andras Zsom

https://github.com/YingfeiHong01/Data1030-FinalProject

Yingfei Hong

1. Introduction

A manager at the bank is disturbed by more and more customers leaving their credit card services. The problem I am trying to solve is to predict the "churned customers" for the bank managers based on the given data so they can proactively go to the customer to provide them better services and turn customers' decisions in the opposite direction. Predicting churn is very important especially when clear customer feedback is absent. Retaining existing customers and thereby increasing their lifetime value is something everyone acknowledges as being important, however, there is little the bank managers can do about customer churn if they don't see it coming in the first place. This is where predicting churn has its value. Early and accurate churn prediction empowers CRM and customer experience teams to be creative and proactive in their engagement with the customer. In fact, by simply reaching out to the customer early enough, 11% of the churn can be avoided [1].

This dataset comes from LEAPS [2] and contains 21 columns and 10127 data points with 'Attrition_Flag' as its target variable in which "Attrited customer" means that this customer is the churned customer we are looking for.

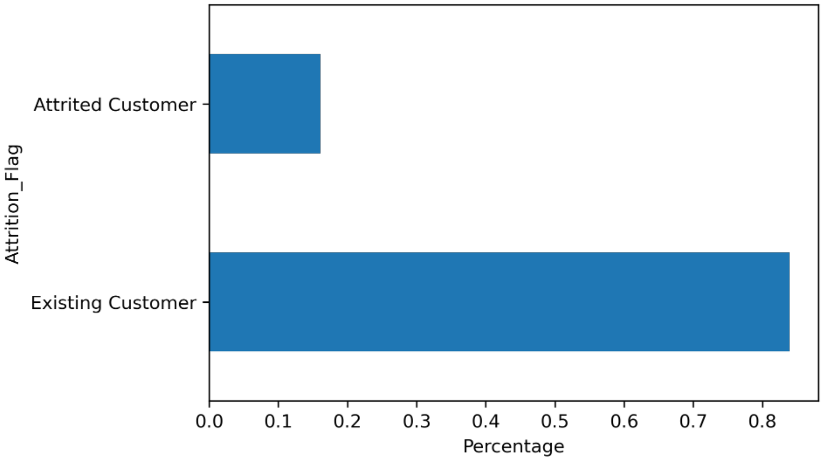

Figure 1 The distribution of 'Attrition_Flag'

It is an imbalanced dataset since the proportion of the 'Attrited customer' is about 16.07%. Besides, through the distribution of the target variable, this task is certainly a typical classification problem.

The rest 20 features are described in the following table which contains their feature name, type, and meaning.

Variable	Type	Description
Clientnum	Num	Client number. Unique identifier for the customer holding the account
Attrition_Flag	char	Internal event (customer activity) variable - flag to indicate if the customer is an existing customer or has attrited
Customer_Age	Num	Demographic variable - Customer's Age in Years
Gender	Char	Demographic variable - M=Male, F=Female
Dependent_count	Num	Demographic variable - Number of dependents
Education_Level	Char	Demographic variable - Educational Qualification of the account holder (example: high school, college graduate, etc.)
Marital_Status	Char	Demographic variable - Married, Single, Unknown
Income_Category	Char	Demographic variable - Annual Income Category of the account holder (< $40K, $40K - 60K, $60K - $80K, $80K-$120K, > $120K, Unknown)
Card_Category	Char	Product Variable - Type of Card (Blue, Silver, Gold, Platinum)
Months_on_book	Num	Months on book (Time of Relationship)
Total_Relationship_Count	Num	Total no. of products held by the customer
Months_Inactive_12_mon	Num	No. of months inactive in the last 12 months
Contacts_Count_12_mon	Num	No. of Contacts in the last 12 months
Credit_Limit	Num	Credit Limit on the Credit Card
Total_Revolving_Bal	Num	Total Revolving Balance on the Credit Card
Avg_Open_To_Buy	Num	Open to Buy Credit Line (Average of last 12 months)
Total_Amt_Chng_Q4_Q1	Num	Change in Transaction Amount (Q4 over Q1)
Total_Trans_Amt	Num	Total Transaction Amount (Last 12 months)
Total_Trans_Ct	Num	Total Transaction Count (Last 12 months)
Total_Ct_Chng_Q4_Q1	Num	Change in Transaction Count (Q4 over Q1)
Avg_Utilization_Ratio	Num	Average Card Utilization Ratio

Table 1 Description for each feature

For this data, there are two main tasks, one is to improve the performance of predicting churned customers while the other is to find the most influential factors that make the customers "churn". However, finding the factors that matter most is a common strategy when enhancing the model performance so I rather treat them as the same task. Several projects have already been done to solve this problem and achieve good results.

Thomas[3] used SMOTE which is an approach to address imbalanced datasets by oversampling the minority class and found great improvement when processing the data generated by this strategy. The result showed that compared to the raw data, this new data could improve the F1 score from an average of 0.6 to an average of 0.9. Andi[4] looked into the details about the raw data and found some interesting relationships between the features and the target variable. The EDA showed that the likelihood of the customers' leaving is related to the money they spend annually, the months of inactivity in their bank account and their credit limit. Joseph[5] used Random Forest and LightGBM to predict with 97% recall and 95% accuracy and plotted the importance of these features which showed that the transaction feature ranked top in both models, so we need to look at these features thoroughly when doing exploratory data analysis.

2. EDA

I've plotted the relationship between every feature and the target variable and found some relations that are worth notice.

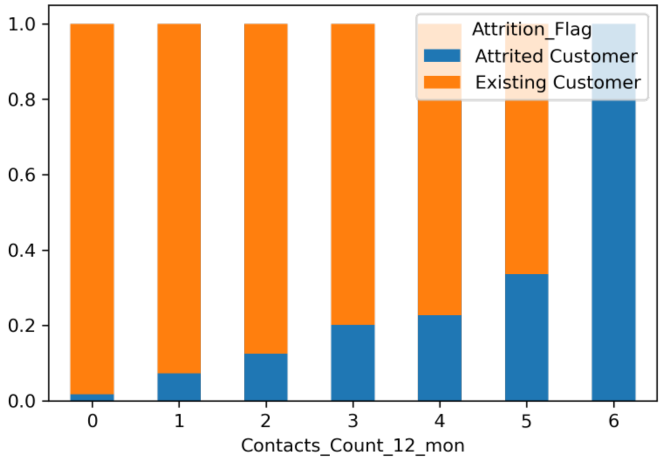

Figure 2 This graph displays how the number of contacts is distributed across two different customers. It seems that churning customers have had more contacts in the last 12 months with the bank managers.

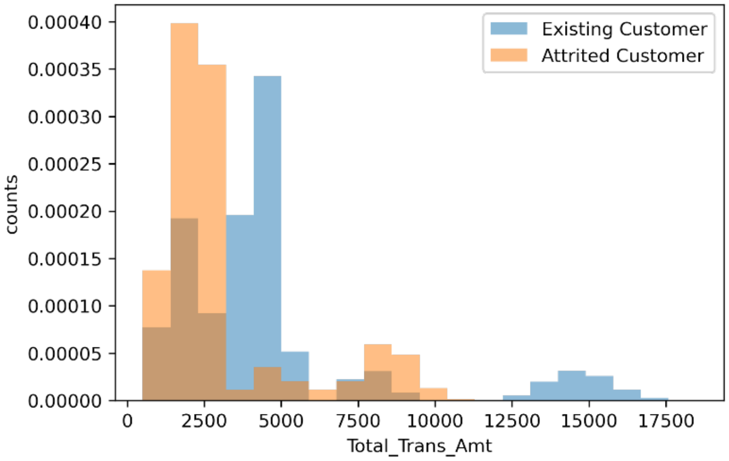

Figure 3 This graph shows that the total transaction amount of attrited customers is smaller than that of existing customers which explains why this feature ranks top in both models of Joseph's projects.

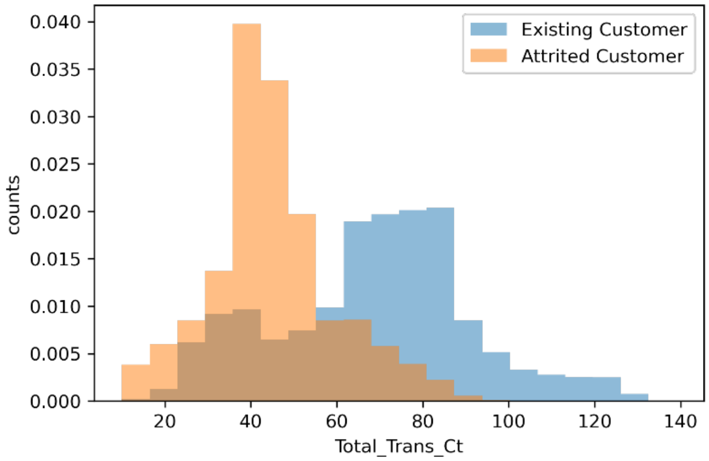

Figure 4 This graph has a similar pattern as figure 3 which shows that total transaction count is also an important feature when we try to classify attrited customers from the existing customer.

3. Method

3.1 Preprocessing

This dataset is not IID since all features are not identically distributed. It does not have a group structure, nor does it have time-series data. But it is imbalanced, so it is better to use stratify method when splitting. The train size is set as 0.6 and the validation size as well as the test size both as 0.2 because it is how people normally split their data when they have lots of data points.

Discrete numerical features (dis_fea) is treated as ordinal features and is put directly into the model since they are already in numerical form. Categorical features (cat_fea) need to be treated with OneHotEncoder because it is not sensible if we put gender and marital status in order. For ordinal features(ord_fea), OrdinalEncoder is the best fit because there is ordinal information contained in educational level, income category, and card category. For continuous numerical feature(con_fea), I chose StandardScaler. Since in these columns, 'Customer_Age', 'Monts_on_book', and 'Total_Trans_Ct' are nearly normally distributed though some are skewed while other features have long tails which are not suitable to use MinMaxScaler.

In this case, I also transform the target variable by setting all "Existing Customer" into 1 and all "Attrited Customer" into 0 to make it machine-comprehensible because it is a binary classification problem and the type of the values in the target variable is ‘string'.

There are some missing values in some demographic variables like educational level, income category, and marital status and we can treat them as one special category since they are all categorical features.

In the end, there are 24 features in total and 6076, 2025 and 2026 data points in training, validation and test set respectively.

3.2 Parameter tuning

The ML pipeline used here is quite simple. For each random state, I first split and preprocess the data as the description above and then run all parameters combinations on the training data and validation data to find the best model in each random state, and calculate the test score based on the best model. In the end, we will have 10 test scores with 10 random states for each model.

In this project, I've tried six different models including three logistic models with different regularization methods and SVM, RandomForest, and XGBoost. The parameters of each model are as the following table.

Model	Hyper Parameter	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	min_child_weight	1, 3, 5, 7
	gamma	0, 0.1, 0.2, 0.3, 0.4
	subsample	0.3, 0.4, 0.5, 0.7, 1
	colsample_by_subtree	0.5, 0.66, 0.75, 1

Table 2 Parameters used for tuning

The evaluation metric chosen here is the f2 score because the loss of treating an attrited customer as the existing customer is far greater than that of predicting an existing customer as an attrited customer. Therefore, we need to pay more attention to recall, and thus the f beta score is chosen as the evaluation metric with large beta as more emphasis on the recall score.

4. Results

4.1 Model evaluation

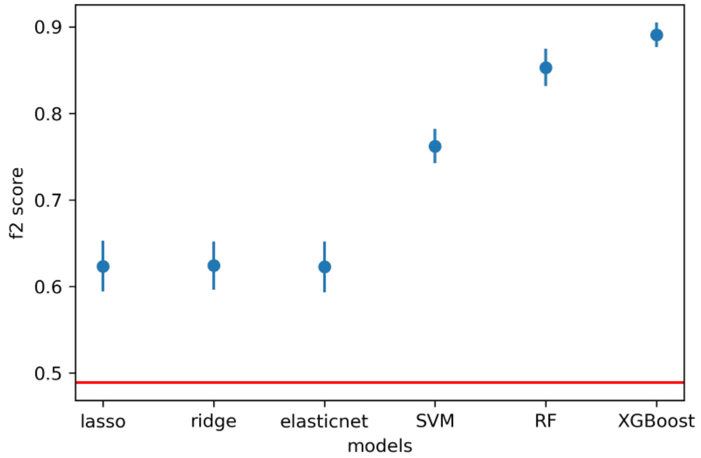

Figure 5 F2-score for the best models over 10 random states

From the errorbar plot above, we can see that all six models achieve better f2 scores than the baseline score. Here the baseline is the model whose predictions are all 1s ("Attrited Customer"). Among these models, XGBoost has the best performance with the best average test score and the lowest variance.

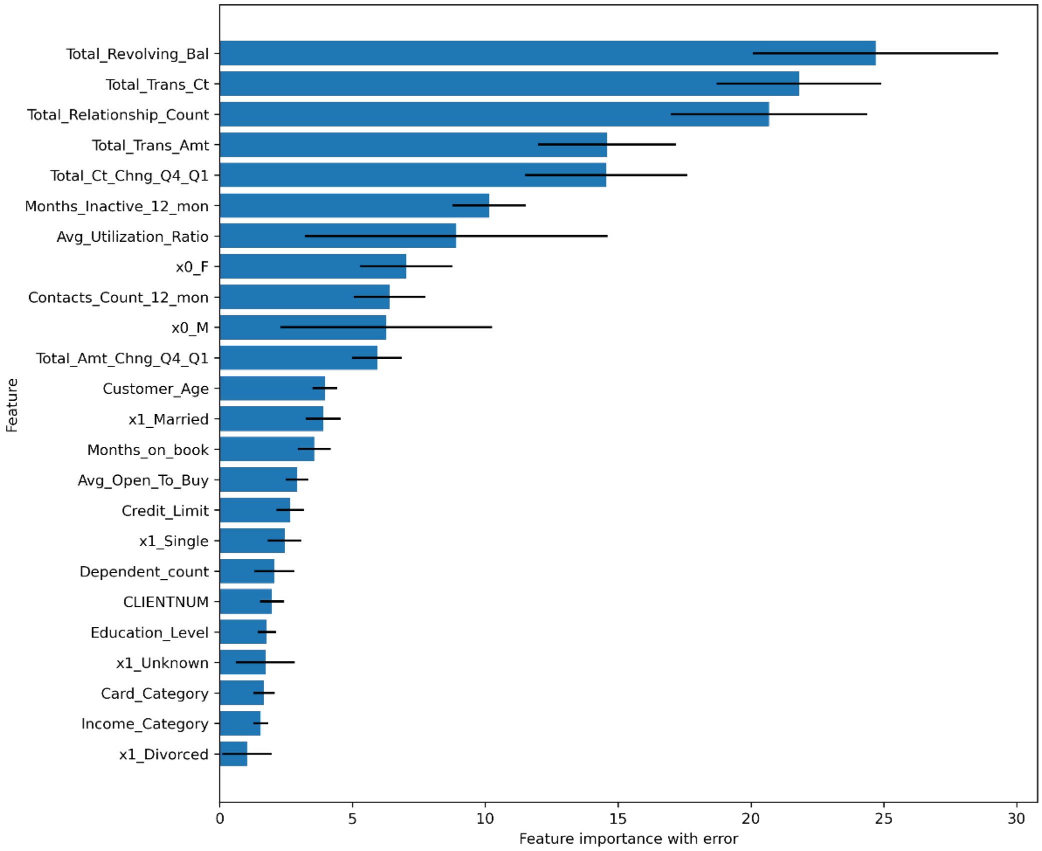

Figure 6 Feature importance with error over 10 best XGBoost models

"Gain" is chosen as the measurement of the importance because it is the most relevant attribute to interpret the relative importance of each feature [6]. The figure shows that the average most important feature of XGBoost models are 'Total_Revolving_Bal', 'Total_Trans_Ct' and 'Total_Relationship_Count. The first two features are related to the customers' activities of bank account while the third one is a demographic feature. And the least important features are ‘x1_Divorced’, ‘Income_Category’ and ‘Card_Category’.

For local feature importance, figure 7 has shown each feature’s contribution to the predictions of two different data points respectively. The base value -1.8 is the average model output over the test data set. The features in red push the prediction higher and those in blue push the prediction lower. And the higher the prediction, the more likely it is an attrited customer. The upper graph shows that the increasing effect of features such as 'Total_Trans_Ct' with value -1.3 is offset by the decreasing effect of features like 'Total_Trans_Amt' with value -0.9. The lower graph shows that 'Total_Trans_Ct', 'Total_Relationship_Count' and 'Total_Amt_Chng_Q4_Q1' with value -1.2, 2.0, -1.8 have the most increasing effect.

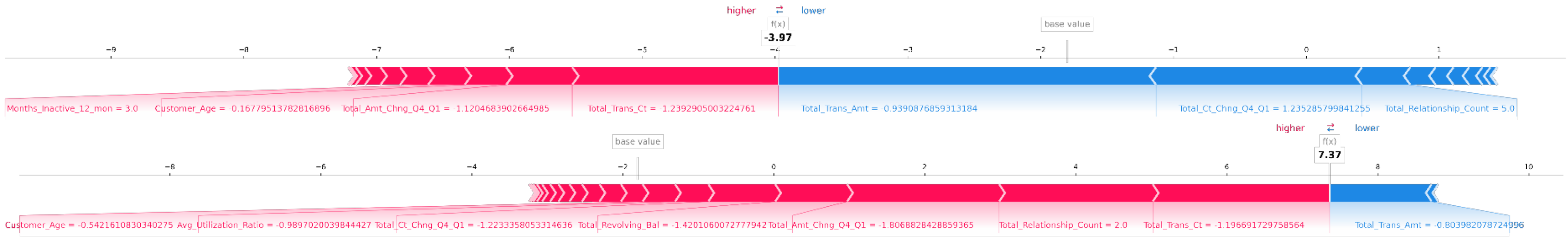

Figure 7 Feature contribution of data points (upper: existing customer; lower: attrited customer)

Besides, there are also some interesting facts about how the feature values affect the model output if we look into the scatter plot of Shap values and feature values.

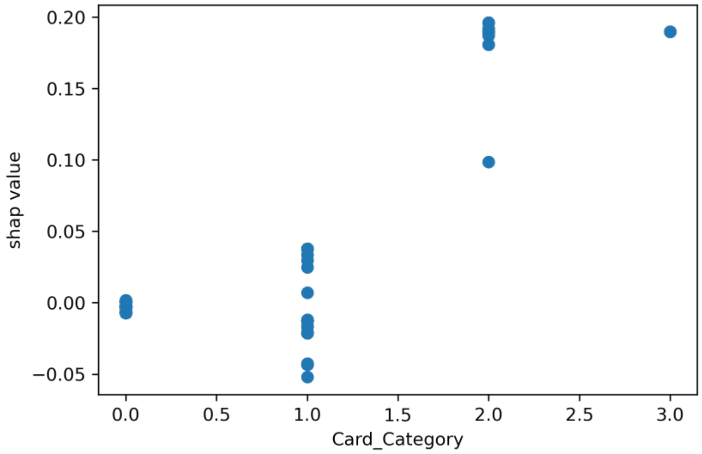

Figure 8 shows that the more valuable the cards customers hold, the more likely they will be predicted as an attrited customer.

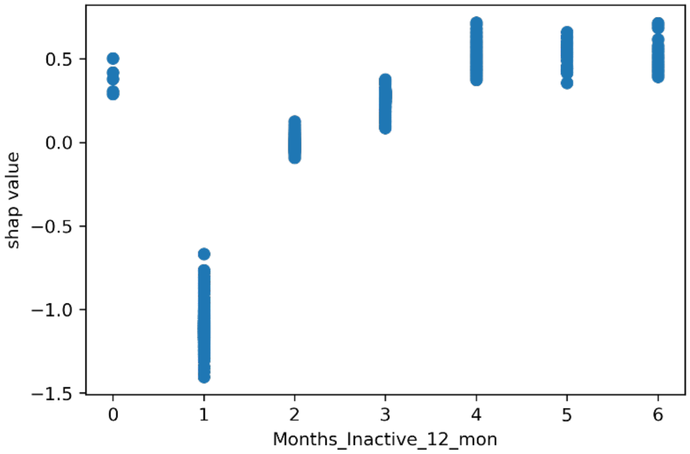

Figure 9 shows that customers will be more likely to be predicted as an attrited customer if they have

more inactive months.

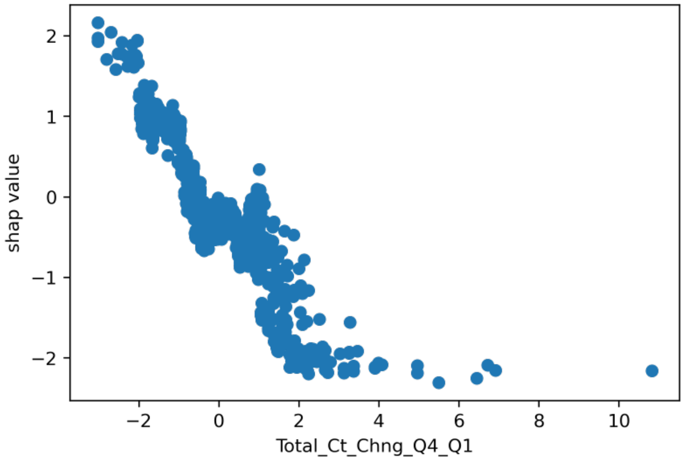

Figure 10 shows that the more changes in total transaction count in O4 over Q1, the less likely the customer is predicted as an attrited customer.

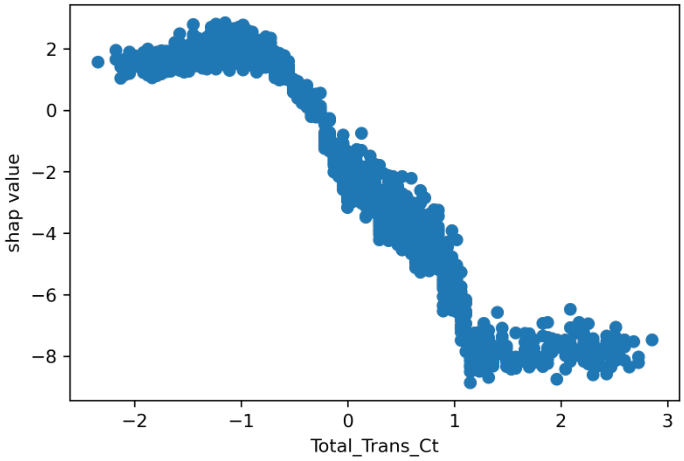

Figure 11 shows that customers who have more total transaction count will be less likely to be predicted as an attrited customer.

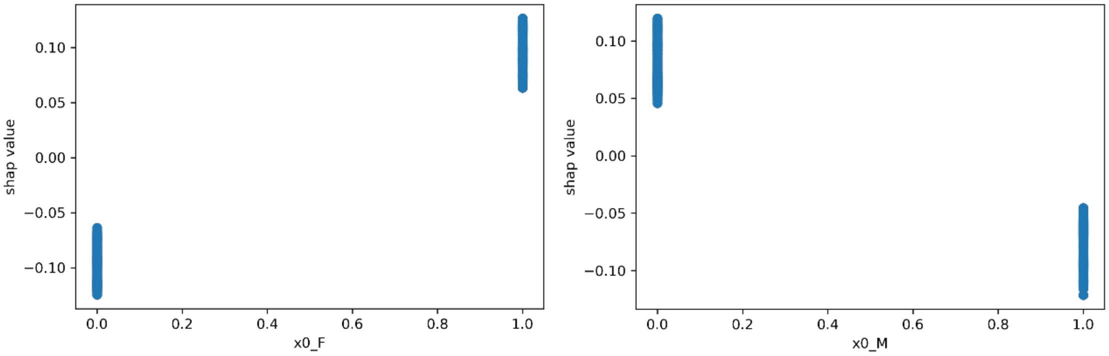

Figure 12 shows the scatter plots of Shap value and the sex feature value. By contrast, we can see that female customers are more likely to be predicted as attrited customers than male customers.

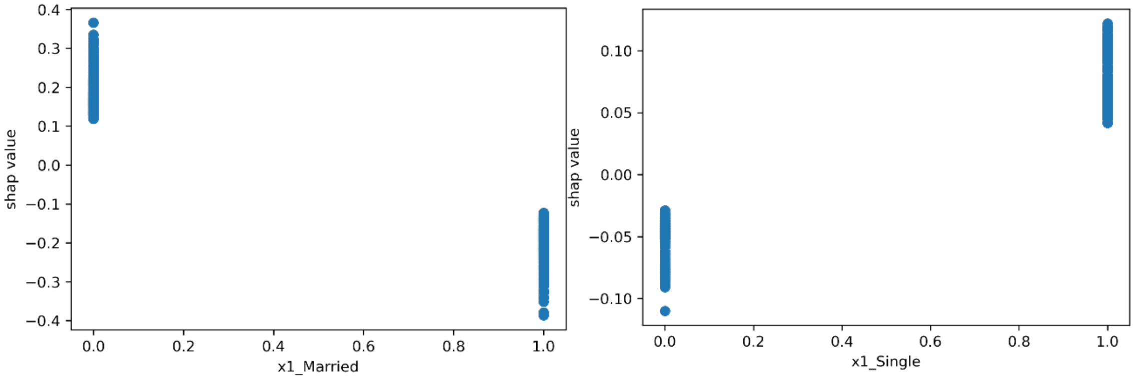

Figure 13 shows the scatter plot of Shap values and the marital status values. Married customers are less likely to be predicted as an attrited customer than single customers.

5. Outlook

One thing that is worth considering is the relationship between two features or among more features for better interpretability. There are some similar patterns in the scatter plot of Shap values and feature values like 'Total_Trans_Ct' and 'Total_Ct_Change_Q4_Q1' or 'Avg_Open_To_Buy' and 'Credit_Limit'. In reality, these features are highly associated, 'Total_Trans_Ct' and 'Total_Ct_Change_Q4_Q1' are both related to the transaction counts while the average amount left in customer's credit card plus the average amount that the customers paid using a credit card is the credit limit of the credit card.

Besides, based on the relationship of 'Avg_Open_To_Buy' and 'Credit_Limit', we can come up with a new feature that represents the average amount that the customers paid using their credit card and this new feature may increase our model prediction. Except for feature engineering, two other strategies may improve the model performance based on previous work. One is to use the LightGBM model because previous work has shown that LightGBM can achieve around 95.7% f2 score. The other is to collect more data of "Attrited Customers" as It has shown that the SMOTE technique which is to oversample the minority class could increase the model performance. The evaluation score also needs more consideration. The actual loss of predicting an attrited customer as an existing customer may be more or less than twice the loss of predicting an existing customer as an attrited customer.

6. Reference

[1] Why customers leave &amp; what can banks do? Tiger Analytics. (2020, September 16). Retrieved October 12, 2021, from https://www.tigeranalytics.com/blog/addressing-customer-churn-in-banking/.

[2] Predict Customer Attrition Using Naïve Bayes Classification. ATH Leaps. Retrieved October 12, 2021, from https://leapsapp.analyttica.com/cases/11.

[3] Konstantin, T. (2021, May 1). Bank churn data exploration and churn prediction. Kaggle. Retrieved October 12, 2021, from https://www.kaggle.com/thomaskonstantin/bank-churn-data-exploration-and-churn-prediction.

[4] IDW, A. (2021, January 31). Customer churn - EDA, 95% ACC and 85% recall. Kaggle. Retrieved October 12, 2021, from https://www.kaggle.com/paotografi/customer-churn-eda-95-acc-and-85-recall.

[5] Chan, J. (2021, January 13). Bank Churners Classifier (Recall: 97% accuracy: 95%). Kaggle. Retrieved October 12, 2021, from https://www.kaggle.com/josephchan524/bankchurnersclassifier-recall-97-accuracy-95.

[6] Abu-Rmileh, A. (2021, September 2). Be careful when interpreting your features importance in xgboost! Medium. Retrieved December 5, 2021, from https://towardsdatascience.com/be-careful-when-interpreting-your-features-importance-in-xgboost-6e16132588e7.