Credit Card Transaction EMI Case Study

Ringan Majumdar, MSc. Statistics

Introduction:

Every bank more or less offer EMI payments facility to their credit card customers. Using EMIs can break down big purchases into smaller parts while skip paying the revolve interest. Now, the bank which issues the credit card tries to have that customer to shift to EMI payment. But, there may be many factors which drives a customer to convert to EMI. This case study is trying find out those customers who has a higher probability of converting to EMI payments.

Data Description:

I am provided with two datasets. First, the "case_study_devdata.csv" which has 50,000 variables and 347 variables. This dataset has one flag variable named "target_variable", which denotes whether the given customer has changed their transactions in EMI or not. The second dataset is "case_study_validation.csv". This dataset has 30,000 variables and 346 variables. This dataset does not contain the flag variable.

Objective:

A procedure is needed such that it gives the probability that any given transaction will be converted to EMI.

Data Handling and Cleaning:

I have faced multiple problems while scrutinizing the data. They are listed as follows.

1. **Variables containing NA values:** There are huge number of variables having NA values. Now, the variables which had more than 30% NA values have been dropped except two. The variables denoting the customers who have opted for EMI before and how many times have they opted for EMI. In these two variables the customers who have not opted for EMI was filled as NA. So, those values where replaced by 0. The remaining variables which has NA values less than 30% has been considered. For these variables, many were continuous and many were discrete. For, discrete variables, the NA values have been imputed with Mode and for continuous variables the NA values have been imputed with the median of the remaining observations of the respective variables.
2. **Variables with only one value:** The variables that takes only one value (such as "Married_flag") has been dropped since they have zero variance and hence will have no contribution on the response variable. Interestingly, all the customers of the development dataset was unmarried.
3. **Data imbalance:** The distribution of the target variable in the development data set is hugely imbalanced. Only 2620 customers have converted to EMI out of 50000 observations. That's about only 5.24%. So, fitting a model to this imbalanced data will automatically give us a high accuracy but low precision. So, over sampling and under sampling techniques have been used to makes this data balanced and then a model is fitted

4. **Variable dependencies or Multicollinearity:** There are some variables which are highly dependent upon some other variables. So, for tackling this issue, the highly correlated variables are first identified then one or two of them are taken by understanding which one is the best.

5. **Huge number of Variables:** There are huge number of variables in the datasets. Moreover, 4 of them are factor variables. The variable merchant name has 18600 levels in the development data set. Hence. not all can be used to fit the model since it becomes overcomplicated and very time consuming.

Model selection:

We are asked to find a way of predicting the probabilities that any given transaction gets converted into EMI. So, we need to use a binary classification algorithm for predicting the probabilities. I have used the logistic regression algorithm as my model to fit the data. Since, it takes much less time than other classification techniques like decision trees, random forests or Neural Networks and when the variables selections are done good it gives very good results. On the other hand, we can assess variable significance of the model, that is, which variable affects the Model output how much.

What is Logistic regression?

Logistic Regression is a type of classification as well as regression technique. When the response variable is a dichotomous variable, that is, it can assume only two values and the predictors are independent factor, discrete or continuous variables one may opt to use logistic regression.

Mathematically, the logarithm of odds ratio is fitted on a linear combination of the predictors,

$$log(\frac{p}{1-p}) = \beta_0 + \beta_1 x_1 + \beta_1 x_1 \cdots + \beta_k x_k$$

where, β_is are regression coefficients, x_is are the predictors and p is defined as $P[Y = 1|X = x]$ that is the probability of the target variable assuming 1 given the predictors.

Variable Selection:

The variables which one can easily understand that will affect the output has been taken in to consideration first. For example,

1. **Transaction amount:** A customer with low transaction amount may try to convert to EMI

2. **Revolve interest rate:** If a customer has high revolve interest rate they might want to convert to EMI to avoid this interest rate.

3. **Credit limit:** A person with high credit limit is probably rich so he might not need or want to use the EMI facilities as much as a person coming from a middle class.

4. **Age of the customer:** Older people tend to avoid EMIs. Interestingly, the age was given as negative valued variable. So, it was converted to positive values.

5. **Credit card product that the customer has opted:** A very rich or a very poor customer is less likely to opt for EMIs than middle class.

6. **Customers who have opted for EMI before:** If someone has opted for EMI before they have relatively higher chance of opting EMI again.

7. **Number of times the customer has opted for EMI:** If someone has opted for EMI multiple times they should have a very high chance for opting EMI again.

8. **Merchant name:** If customers opt for EMI specifically for some selected merchants they should be of some importance. But, there are 18600 unique merchants in the development dataset. We can use all of it. So, I have classified the merchant names in to two groups. First, I have obtained the which merchants have the highest number of EMI conversion. Then, I took the first 10 merchants having highest number of EMI conversion as a class and the remaining another class. This is how, I have made a new variable using the merchant name having two classes.

9. **Merchant country:** It would be interesting to have whether a country have high EMI conversion rate or not. But, here too there are huge number of countries. So, I have made a new variable having two classes. One class denotes top 10 countries having highest EMI conversion and the other class being the remaining.

10. **Merchant category:** The merchant category is a hugely important variable, because we have biases in using EMI for different sector. For example, we don't use EMI to buy train tickets or food but many people opt for EMI when buying Electronics or paying Insurances. But, in this case too, there are huge number of categories, so we have taken the top 10 categories as class and the remaining as another, making another new variable.

11. **Customer spending in the last 1, 3, 6 or 12 months:** This spending habit might also affect the EMI conversions. We have only used the variable spends_12m because all these 4 variables are highly correlated.

12. **Average utilization in last 1, 3, 6 months:** How much a credit card is utilized is an important factor. Now, util_1m, util_3m and util_6m are highly correlated so we have only used util_6m.

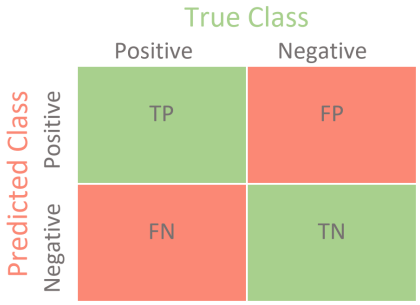
13. **Average ratio of repayments to total outstanding in the last 1, 3, 6 months:** If a customer can repay total outstanding within time he/she might opt for EMI. Only using payment_ratio_6m since all three payment ratios are correlated.

14. **Average ratio of repayments to minimum amount in the last 1, 3, 6 months:** If a customer can repay minimum amount within time he/she might opt for EMI. Here, we are using paymad_1m and paymad_6m, since paymad_1m and paymad_3m are correlated but paymad_6m is not.

15. **Rupee transaction value and number of transactions of different merchant categories:** Only selected those merchant categories by which have the top 7 EMI conversion numbers. And even on those I have taken transactions in last 1 and 3 months since 6 and 12 months' transactions were correlated with the other two. Same goes for the number of transactions. The top merchant categories with highest conversion numbers are Hotels, Electronics, Insurance, Fuel, Rent, Department stores, Utility.

Some Metrics:

1. **Confusion Matrix:** This is a type of matrix which is used to analyse the performance of a classification algorithm. It is defined as follows,



where, TP is true positive(predicted true and its true), TN is true negative (predicted negative and its true), FP is false positive (predicted true and its false) and FN is false negative (predicted false and its false).

2. **Accuracy:** This is defined as, $\frac{TP+TN}{TP+TN+FP+FN}$. In other words, it is the proportion of correctly classified data points among all data points.
3. **Precision:** This is defined as, $\frac{TP}{TP+FP}$. It is actually the proportion of positives among all predicted positives.
4. **Recall:** This is defined as, $\frac{TP}{TP+FN}$. In other words, this measures how many of the positives were predicted correctly in the full positive class.
5. **F1 score:** This is defined as, $\frac{2\times Precision\times Recall}{Precision+Recall}$.

Data imbalance:

The target variable in the development dataset is highly imbalanced, that is, only 5% of the customers have converted in to EMIs. Hence, fitting the model on this data was giving very high accuracy, around 94%. But the precision, recall and the F1 score was very low in this case, very close to zero.

Hence, I have used SMOTE for balancing the target variable using the DMwR library in R. SMOTE stands for Synthetic Minority Oversampling Technique. It is a technique by which synthetic samples are generated for the minority class, overcoming the overfitting problem of the data.

Using this technique, I have balanced the target variable data in around 1:1 ratio. Then, the model is fitted on the new validation data.

Splitting the development data set into train and test:

The development dataset is split into two subsets, one for training and another for testing of the model. The split has been in done in a 70:30 ratios. 70% of observations have been randomly sampled from the development data to make the training set and the remaining observations go to the testing set.

Model results for training and testing set:

The model results for training data is significantly good now. The values of accuracy, precision, recall and F1 score came out as 88.7%, 83.0%, 84.8% and 0.839 respectively.

Although, the values of accuracy, precision, recall and F1 score for the test dataset came out as 89.9%, 49.6%, 24.4% and 0.327 respectively. This is significantly higher than what it was when SMOTE was not used to fix the data imbalance. The precision is now more or less 50% whereas it was very close to zero when SMOTE was not used. This means, this model has predicted around 50% of the customers customers converting to EMI, which is a major improvement.

Model fitting of the development dataset:

The above logistic regression model is now fitted over the whole development dataset. More or less all the variables selected for the model had a significant effect on the output. Moreover, the values of accuracy, precision, recall and F1 score came out as 88.7%, 82.8%, 84.9% and 0.838 respectively. The metrics are very close to what we obtained for training dataset.

Output

The model is then used to predict the probabilites from the validation data set. The predicted probabilities for the validation dataset along with the primary keys is attached to the google drive link as a csv file.

Conclusion

It is clearly observed that there is a huge imbalance of the target variable in the development data set. Hence, a fit to that data set is producing very high accuracy but precision, recall or F1 score was very close to zero. Hence, this model was not able to identify the customers who were more likely to convert into EMI at at all. Now, one can understand that misclassifying a customer who is more willing to convert to EMI as a customer who does not want to convert to EMI is much more problematic than misclassifying a customer who is not willing to convert to EMI as a customer who actually wants to convert to EMI. Since, if somebody wants converts to EMI and the Bank does not know that, then, the bank might lose a probable EMI convert. But, if the bank invests time, effort and money on somebody who doesn't really want EMI, the customer might get convinced in some cases where despite him not wanting EMI, he will go for it, which will be good for banks perspective.

Our new model helps to achieve that, despite having really low precision, recall or F1 score it is far better that what it was previously.