IT1244 Project — Team 11

Written by Beh Qian Jun, Clement Ling, Hor Zhu Ming, Tan Tze Heng, and Than Hui Xin

[1]National University of Singapore

Introduction

Over 1 billion credit card transactions are made globally every single day(1). With the plethora of advantages credit cards offer — convenience, rewards, establishing credit, and avoiding fraudulent charges — they are indisputably a paramount instrument that has permeated modernity, and facilitated the flow of our economy.

Foreseeably, this alludes to an exponential number of credit card applications for commercial banks to filter through. To illustrate, in 2022, credit card applications at UOB saw a spike of 44%(2). However, manually evaluating these applications is error-prone, and expensive in both time and resources. Commercial banks, hence, capitalised on breakthroughs in AI to automate the credit card approval process using methods such as Simple Linear Regression(3).

However, the dataset we chose is imbalanced due to having considerably more observations of "good" clients than "bad" clients. We are unaware of how these ML techniques employed by Commercial banks will perform on such skewed data. Therefore, to find out the performance of these models, and aid banks with smaller datasets in their ML endeavours, our group aims to apply the ML techniques learnt in IT1244 on automating the credit card approval process.

We will first preprocess the data and establish the credit scores to aid our Machine Learning (ML) model in objectively quantifying the magnitude of risk for issuing credit cards to a particular consumer. This is followed by the implementation of several ML models — Simple Linear Regression, Polynomial Linear Regression, Logistic Regression and Random Forests, where the latter three were employed due to the poor accuracy acquired by the Linear Regression model.

Dataset and Methodology

In this project, the credit card approval dataset was used, which consists of the application.csv and credit_record.csv. The application.csv comprises of attributes which are the personal information of clients, and the credit_record.csv contains the credit status of clients for the current and the past months.

Preprocessing of Data

After understanding the two CSV files that constitute the dataset — the credit records and applicants' particulars files, we started with data preprocessing. The "job" variable consisted of a significant number of NULL entries, and the entries were inadequate in determining the role the job type had on credit scores. Hence, the "job" column was removed to facilitate easier processing of our data. Subsequently, the Pandas Package was employed to remove any other existing NaN entries. Furthermore, the values for "own_car" and "own_realty" were converted to numerical data to enable the ML models to be able to process the data; "Y" and "N" were switched to 1 and 0 respectively.

Any illogical data was checked and removed, such as entries where there was negative income, employment lengths that exceeded the number of days since birth, and when the number of family members was fewer than the number of children. Since the employment lengths were calculated backwards, negative employment lengths signified employment, while positive lengths indicated that these individuals were unemployed. Therefore, to simplify the data, the employment lengths of employed individuals were converted to positive values, while the employment lengths of unemployed individuals were all converted to 0. Birth_day was also converted from negative to positive values.

Interpretation & Definition of Credit Score

The credit records CSV file constitutes the statuses of the applicants, such as whether they had loan for the month, if they paid the loan on time, or how overdue the payment was. We then utilise these pieces of information to compute the credit score for every individual.

Many online sources and research papers define the credit score of customers, $C(customer)$ to be a discrete variable.

$$C(\text{CLIENT}) = \begin{cases} 1, & \text{NO PAST OVERDUE} \\ 0, & \text{HAVE PAST OVERDUE} \end{cases} \quad (1)$$

Our group interpreted the credit score $C(\text{CLIENT})_{our}$ to be a continuous spectrum number between $[0, 1]$ given by the following equation:-

$$C(\text{CLIENT})_{our} = \frac{\text{SUM(ALL HISTORY SCORE)}}{\text{SUM(MAX POSSIBLE SCORE)}} \quad (2)$$

RECORD	C	0	1	X	2	3	4	5
SCORES	10	8	7	5	1	1	0	0

Table 1: Example of credit score conversion.

Afterwards, an inner join was executed on the credit records and applicants' particulars CSV files to match the IDs of the applicants together, where the computed credit scores were the class labels of each applicant.

With that, we embarked on the implementation of the ML techniques.

Results and Discussion

Attempt 1: Simple Linear Regression Model

Firstly, a Simple Linear Regression model with stochastic gradient descent (SGD) was employed to predict the credit score of the clients. Although the calculated mean squared error (MSE) and the mean absolute error (MAE) were low (0.00859 & 0.1681 respectively), the model seemed to be overfitting, specifically to the class of "good" clients, who we defined as applicants with a credit score ≥ 0.6. This is due to the highly imbalanced dataset, in which only around 100 of the 30,000 clients were defined as "bad" clients.

Hence, we decided to replicate the client with the lowest credit score 8,000 times in hopes of making the dataset more balanced. However, the predicted result was evidently still overfitting in favour of the "good" clients and the values of MSE and MAE still appeared relatively constant.

Attempt 2: Polynomial Linear Regression & Logistic Regression

Next, we decided to employ both polynomial linear regression and logistic regression to predict the credit scores. Although the polynomial linear regression model appeared to have a better fit as compared to the simple linear regression model, the resulting loss of the model still seemed to be sufficiently low when the test data was applied to the model.

As an effort of trial-and-error, binary class values ('good' or 'bad') were also explored instead of a continuous value of credit score. This meant that a logistic regression model could be employed.

Defining a threshold value of 0.5, we assigned rows with credit scores above that threshold to take on the class value ('score') of 1, and 0 otherwise. We then fit the training data into the logistic regression model.

The accuracy of the logistic regression model is $\approx 99\%$, which is sufficient for a logistic regression model. However, the problem of overfitting in favour of the "good" clients still remains unresolved.

Attempt 3: Random Forest Model

From prior attempts, we noticed that overfitting is the underlying cause of the model being unreliable. Therefore, we chose a model more resistant to the conundrum of overfitting — the Random Forest Model. Compared to Decision Trees, the Random Forest Model is significantly less sensitive to outliers, and much more resistant to overfitting, hence our decision to utilise the model over Decision trees.

Random Forest is essentially a model that constitutes multiple Decision Trees, where each individual tree chooses a random combination of variables to form the decision tree. The output is thus the class selected by most trees. The model overcomes overfitting by applying the technique of bootstrap aggregating, or bagging, to the trees in the model and the random variables selected by each individual tree, resulting in a model more resistant to outliers.

As a result, the accuracy of the Random Forest model in predicting the credit scores is $\approx 92.3\%$, by calculating the Mean Absolute Percentage Error (MAPE), in which MAPE is defined as $100 \times \frac{|y_{pred} - y_{test}|}{y_{test}}$. After integrating the duplicated data into the dataset, the model can now predict credit scores ranging from 0 to 1, reducing the predominant issue of overfitting.

Conclusion and Future Work

Overall, the Random Forest Model appears to be the best-performing model as it best overcomes the limitation of an imbalanced dataset.

As an effort for future work, Synthetic Minority Oversampling Technique (SMOTE) could be implemented to overcome the problem of imbalanced datasets. However, a noteworthy limitation is that the accuracy of SMOTE generally declines as the dimensions of the data increase. Thus it is unsuitable for non-linear data.

Moreover, there would also be a huge leap in the performance of the model if the past history of clients' behaviours were included as one of the variables in the dataset, as this would further increase the correlation of the variables, and hence fit of the model.

References

[1] Average Number of Credit Card Transactions Per Day Year. https://www.cardrates.com/advice/number-of-credit-card-transactions-per-day-year/ (accessed 2022-10-31).

[2] Banks see spike in credit card applications as Covid-19 restrictions taper off, Banking Finance - THE BUSINESS TIMES. https://www.businesstimes.com.sg/banking-finance/banks-see-spike-in-credit-card-applications-as-covid-19-restrictions-taper-off (accessed 2022-10-31).

[3] Singh, R. Predicting Credit Card Approvals using ML Techniques. Medium. https://medium.datadriveninvestor.com/predicting-credit-card-approvals-using-ml-techniques-9cd8eaeb5b8c (accessed 2022-10-31).