Default Payment of Credit Card Clients

1. Introduction

Credit card issuers have become one of the major consumer lending products worldwide, representing more than a third of total consumer lending in the United States. For many customers, credit cards are a flexible tool by which they can buy the product they want first and agree to pay their bills later by the due date according to the credit card statement.

However, credit card lending is highly risky for card issuers because such loans are not secured by any assets. When people fail to make this payment by the due date, their credit card is defaulted, causing significant financial losses for the company. Due to limited subsequent checks of financial status following the initial issuance of the card, credit cards often elicit endless waves of borrowing, with only the borrowers themselves knowing their ability and willingness to actually repay. Given such circumstances, the inability to accurately and consistently predict which customers are likely to not pay back the debt is a significant business problem for many companies. This project aims to model the potential risk a customer poses in order for credit card companies to identify and prevent predictable damages.

Credit card issuers will use this model to interpret the historical information of customers' accounts, using past transaction information to predict which customers are likely to not make the payment. In doing so the credit card company will be able to identify and distinguish the 'good' or 'low-risk' customers from the 'bad' or 'high-risk' customers. After the 'bad' accounts have been identified, additional steps and investigations will be taken by the company to verify the actual nature of the accounts.

By the analysis above, we develop our analysis by following the four steps:

First, we analysis the data set and clean the data by removing outliers and normalize numerical data.
Then we design machine learning models with Python and determine the parameter for each model by complexity control and observing fitting graphs.
Evaluating each model and analysis which model should be employed under certain business situation.
Finally, discuss what the company need to notice while using the model provided and how the result of data mining can be deployed.
\end{enumerate}

2. Literature review & Motivation

In July 2017, Morten Hansen Flood from the Norwegian University of Science and Technology conducted a research project for early identification of high-risk credit card customers based on behavioral data. The research aimed to determine whether it was possible to identify high-risk customers within the first few months of their joining. By using a small dataset consisting of only the first two months of customers' data, several machine learning methods were applied to develop classifiers that attempted to predict future delinquency.

Apart from predicting delinquency, several models aimed to analyze customer behavior that drove delinquency and to model credit risk. The results showed that it was very difficult for models to accurately predict high-risk customers with such limited data consisting of only the first few months. The factors that drove delinquency have also turned out to be primarily intuitive. Without analyzing any information regarding the customers' spending patterns, it was even more difficult to distinguish customers who had no intention of paying their debt from the customers who were actually going to pay.

The study by Morten Hansen Flood used the first two months of the customer relationship in order to accurately predict which customers will default in the following months. In order to develop a more accurate model to predict high-risk customers, our research incorporated a larger dataset spanning for a time period of six months. By accurately identifying low-risk

customers and high-risk customers, we aim to minimize the financial damages for credit card issuers due to 'bad customers.' Our model will prevent costs from situations resulting from inaccurate predictions, such as having to pay for a 'bad' customer's bankruptcy, or costly investigations and premature bans on 'good' customer accounts.

3. Data Overview

In this project, we will examine the specific pattern of "bad" credit card account on a rather precise level, in other words, the features of default payment. Since historical data are being applied, the analysis will be a supervised learning process, and more specifically, classification. This analysis employs a binary variable, default payment (Yes=1, No=0), as the target variable.

As presented in the Data Source part above, our possible features, based on the data we have, include personal information about users and individual payment history. For example, the user's *gender*, *education level*, *marital status*, and *age* are captured as personal information features. User's *amount of given credit*, *history of payment*, *amount of bill statement* and *amount of previous payment* are used as features regarding individual payment history. We also plan to employ their interactions, if necessary, to improve the completeness of the model.

We found a dataset on the Machine Learning Repository of University of California Irvine, which contains 30000 instances in total.
URL: https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients#

3.2 Introduction of data

3.2.1 Variables

I think variable introduction

To provide an answer to the prediction, we will use whether the user will default the payment next month as our target variable and will use other variables in dataset as independent variables and features. And we also dropped feature 'ID' in the analysis since the feature itself is meaningless for analysis.

- Target variable:

- Default_payment_next_month

A dummy variable indicating whether the customer will default the payment next month. (1 = will default, 0 = will not default).

- Control variables:

- AL – the amount of the given credits.

It includes both the individual consumer credits and his/her family's (supplementary) credits (I think the result is the sum of his individual credits and his/her family's credits --- **need further discussion**). From intuition, we think people with higher total credits will have lower probability to default the payment next month.

- SEX – Gender of the customer (1 = male; 2 = female).

This variable is used to check whether the default on payment is gender-related.

- EDUCATION – Education level of the customer (1 = graduate school; 2 = university; 3 = high school; 4 = others).

Intuitively, we think people with higher education level will have lower probability on defaulting payment.

- MARRIAGE – Marital status of the customer (1 = married; 2 = single; 3 = others). The variable is used to check whether the default on payment is related with the customer's marital status.
- AGE – Age of the customer.

This variable is used to check whether the default on payment is age-related.

To track customer's payment history, we include 6 variables to indicate customer's historical repayment status, from September to April, 2005. In our dataset, not every customer start in the same month, i.e. start in September and end in April. Therefore, we will have the value equal to 0, for the months that the customer have not yet joined.

- **PAY_9** – Customer's repayment status in September 2005 (-1, 1, 2, …, 9)

-1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

- PAY_8– Customer's repayment status in August 2005 (-1, 1, 2, …, 9)

-1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

- PAY_7– Customer's repayment status in July 2005 (-1, 1, 2, …, 9)

-1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

- PAY_6– Customer's repayment status in June 2005 (-1, 1, 2, …, 9)

-1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

- PAY_5– Customer's repayment status in May 2005 (-1, 1, 2, …, 9)

-1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

- PAY_4– Customer's repayment status in April 2005 (-1, 1, 2, …, 9)

-1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

In order to analyze the probability of default repayment, we include the amount of customer's history bill statements for six months, from September to April, 2005. In our dataset, not every customer start in the same month, i.e. start in September and end in April. Therefore, we will have the value equal to 0, for the months that the customer have not yet joined.

BILL_AMT9 means the amount of customer's bill statement in September 2005, BILL_AMT8 means the amount of customer's bill statement in August 2005, etc.

- BILL_AMT9 – Amount of bill statement in September 2005 (in New Taiwan dollar).
- BILL_AMT8 – Amount of bill statement in August 2005 (in New Taiwan dollar).
- BILL_AMT7 – Amount of bill statement in July 2005 (in New Taiwan dollar).
- BILL_AMT6 – Amount of bill statement in June 2005 (in New Taiwan dollar).
- BILL_AMT5 – Amount of bill statement in May 2005 (in New Taiwan dollar).
- BILL_AMT4 – Amount of bill statement in April 2005 (in New Taiwan dollar).

Again, In order to analyze the probability of default repayment, we include the amount of customer's historical payment for six months, from September to April, 2005. For convenience, we rename the variable according to the month it tracked, i.e. PAY _AMT9 means the amount of customer's historical payment in September 2005, PAY _AMT8 means the amount of customer's historical payment in August 2005, etc.

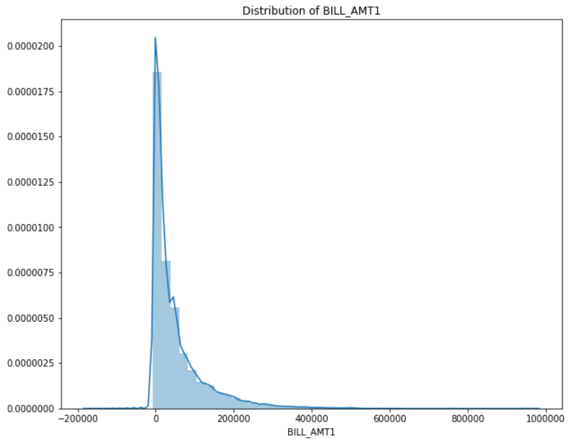
- PAY_AMT9 – Amount of historical payment in September 2005 (in New Taiwan dollar).
- PAY_AMT8 – Amount of historical payment in August 2005 (in New Taiwan dollar).
- PAY_AMT7 – Amount of historical payment in July 2005 (in New Taiwan dollar).
- PAY_AMT6 – Amount of historical payment in June 2005 (in New Taiwan dollar).
- PAY_AMT5 – Amount of historical payment in May 2005 (in New Taiwan dollar).
- PAY_AMT4 – Amount of historical payment in April 2005 (in New Taiwan dollar).

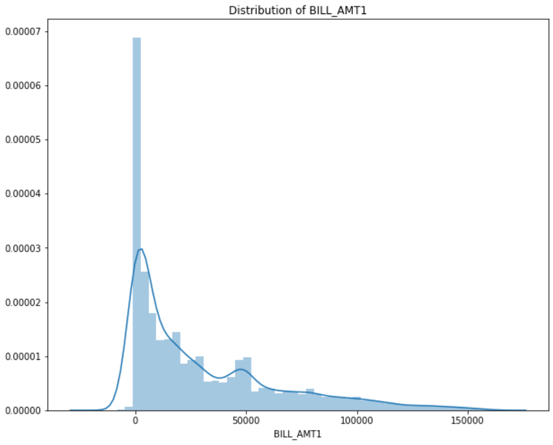
prof comment:

Since the difference between the bill statement and the paid amount represents how much the customer have not pay given a certain bill amount, which indicates the customer's willingness and ability to pay back the account. So we add features DIFF_1, .., DIFF_6, by subtracting the bill statement (BILL_AMT) with historical payment in that month(PAY_AMT).

3.2.2 Data Cleaning

Requirement: Specify how these data are integrated to produce the format required for data mining. (Note: data preparation can be time consuming! Get started early. Talk to the CAs or Prof





Outlier is an observation that lies outside most of the other observations in a data set. Features with huge numerical range are very easy to cause problems in modelling steps and models will be negatively influenced by the outliers. Features regarding bill amount (BILL_AMT) spread very large range. As shown in the graph(ref figure 1), the distribution of BILL_AMT1 has long tails in both sides and the tails need to be removed. However, outlier detection is a complex topic, involving the trade-off between reducing the number of observations and having outliers skew the result of the predictions. We employ the method of interquartile range. The interquartile range is often used to detect the outliers in a data set. Outliers are defined as observations that fall below Q1 − 1.5 IQR or above Q3 + 1.5 IQR. In our case, we only remove the instances that fall in the removal range from BILL_AMT1 to BILL_AMT6. After remove the outliers, the distribution of BILL_AMT1 is presented in figure 2, which is less skewed. In all, we reduced the data size from 30000 transactions by 3767 transactions to 26233 transactions.

After removed outliers, we normalized all the numerical data. Normalizing means to transform the data into a Gaussian distribution by subtracting the mean and by dividing by the standard deviation. After being normalized, every feature falls in the range of [0,1].

4 Data Mining Model

We have 7 models in all, which should all be mentioned

-Primary Model: Decision Tree / Logistic Regression. For the decision tree model, we will learn from the fitting graph and choose tree depth at "sweet spot". We will also determine the best minimum leaf size by analyzing the learning curves to avoid overfitting as well as to get the highest accuracy. For the logistic regression, we will apply the regularization parameter to reduce the sensitivity of the model.

- Resampling Techniques:

1) Percentage Split: train_test_split

We split the data-set into training and test set. This helps us train the model on unseen data, and then use the rest of the data for testing our model.

2) K-fold cross-validation

Machine learning offers a high level of modeling freedom, it tends to overfit the data.

A model overfits when it performs well on the training data but does not perform well on the validation data. K-cross validation helps to minimize overfitting. In k cross-validation approach the data is split into k equal folds. 1 fold is converted into validation set and (k-1) left folds are then used as training set. This goes through k iterations. And each time out of that k folds, 1 is held out as validation set and rest as training. k-folds also helps in smoothing out noisy or

random data. Also, this way we are able to utilize the whole data-set as training, and we are able to test on the whole data-set as well. This helps in making the model as well rounded, which is not just biased towards particular training or validation set. Generally, we'll use k=10.

-To improve the overall accuracy, or completeness, we'll also explore more complicated models as Random Forest to see the performance. In addition, we are also considering, if possible, doing a **hybrid model** to boost accuracy.

5 Evaluation

Requirement: Discuss how the result of the data mining is/should be evaluated. How should a business case be developed to project expected improvement? ROI? If this is impossible/very difficult, explain why and identify viable alternatives.

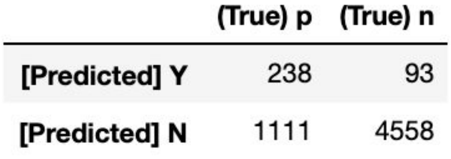
Since the non-fraud observations are the majority of the data set, predicting every instances to be non-fraud instance gives a relatively high accuracy result. Also, in real business situation, predicting a true fraud data correctly is the main aim of the task. As a result, to evaluate model performances and decide which model should be adopted in the real application, this report focuses on precision, Fβ score, and the AUC metric.

Precision is the ratio of the number of true positives by all of the actually correct data points (true positives + false positives). It can be seen as a measure of the quality of the data returned as positive. The equation for precision is: precision =true positives/(true positives + false positives).

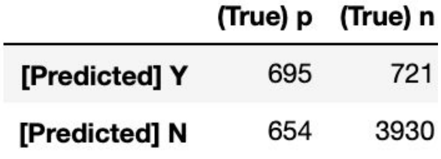
Recall is the ratio of the number of correct positive predictions to all actual positive observations. The recall checks the percentage of true positives were predicted as positive. Recall is also refered as the sensitivity of the model. The equation for recall is: recall =true positives/true positives + false negatives.

The Fβ score can be interpreted as the weighted harmonic mean of precision and recall. Fβ reaches its best value at 1 and its worst value at 0. The beta parameter determines the weight of precision in the combined score. $\beta < 1$ lends more weight to precision, while $\beta > 1$ favors recall. In our model, we choose $\beta = 0.5$ to give more weights to the precision of the model. The equation for Fβ score is: Fβ = (1 + β^2) *(precision * recall)/(β^2 + precision) + recall.

During the evaluation of recall, the recall for logistic regression is 0.034, which means that the number of false negatives are much more than true positives. To better study this problem, we output the confusion matrix (figure 3), which shows that with 238 true positive instances, there are 1111 instances are false negatives. Hence the threshold of being set to be positive need to be adjusted. The default threshold is 0.5 and after plotting the probability distribution, we change the threshold into 0.25, and with this threshold, new confusion matrix is get. The number of true positive observations increases while the number of false negative decreases. With new threshold, the recall of logistic regression model becomes 0.537.



	(True) p	(True) n
[Predicted] Y	238	93
[Predicted] N	1111	4558



	(True) p	(True) n
[Predicted] Y	695	721
[Predicted] N	654	3930

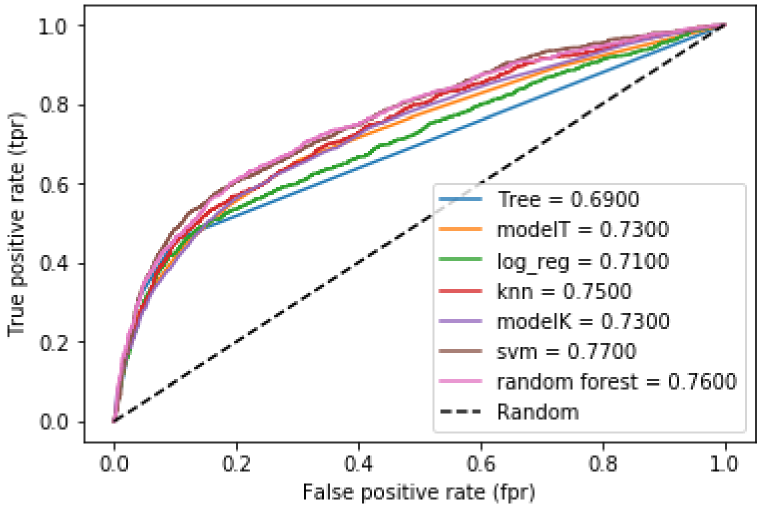
The precision, recall and Fβ regarding the optimized complexity parameters of each model is presented in the following table:

Model	Precision	Recall	Fβ (β=0.5)
Model T	0.588	0.347	0.520
Logistic Regression	0.661	0.537	0.446
Model K	0.626	0.252	0.489
Random Forest	0.684	0.265	0.524
Decision Tree	0.705	0.302	**0.548**
K-NN	0.647	0.273	0.510

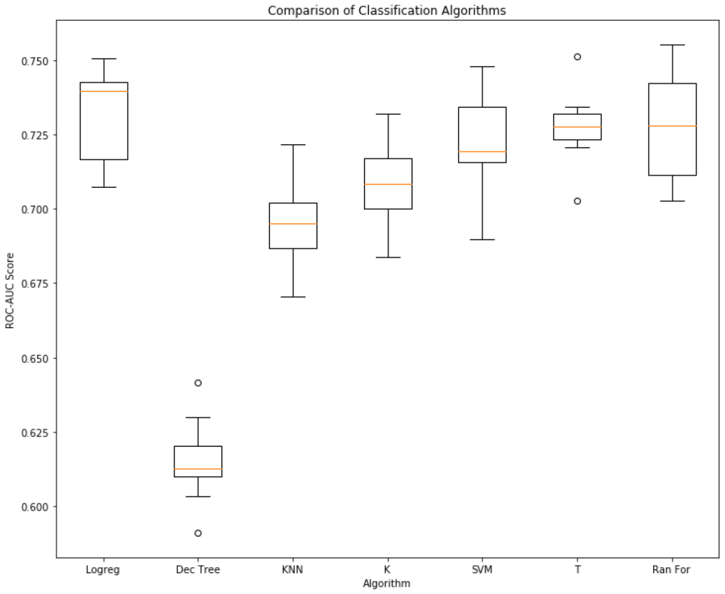
SVM	0.518	0.536	0.531

From the table, Decision Tree model with maximum depth being 2 and minimum sample leaves being 15 outperformed other models. Random forest is the next and logistic regression performs the poorest. We believe that this is due to the fact that logistic regression is based on objective function minimization and tries to capture all features at the same time. Whereas the Decision Tree and Random Forest model focus on most significant splitter in input variables each time.

We also used summary statistic -- the area under the ROC curve (AUC) to help with the evaluation. As the name implies, this is simply the area under a classifier's curve expressed as a fraction of the unit square. Its value ranges from zero to one. A value of 0.5 corresponds to randomness (the classifier cannot distinguish at all between positives and negatives) and a value of one means that it is perfect in distinguishing them. [cite textbook] The AUC graph is presented below:



From the graph(ref graph auc), we can see that all of the 7 models perform better than random guessing. The SVM model performs the best and the next is random forest model.
Finally, we employ spot-checking algorithm and visualize the result regarding 7 models. Spot-checking algorithms is about getting a quick assessment of several different algorithms on the machine learning models so that we can decide which algorithms to focus on and which one to discard. The visualization is presented below:



In the spot checking algorithm, the logistic regression and random forest model outperform the others.

To get a more comprehensive evaluation, we combine the above statistics and find that random forest model performs well in each of the statistic values. Also, considering this project had not only the focus of achieving the best performance of the model but also to create business value, choosing Random Forest is a reasonable approach in order to achieve a higher degree of comprehensiveness while slightly decreasing performance in some way. As a model based on Decision Tree, the output is very easy to understand for people from various non-analytical background. It does not require any statistical knowledge to read and interpret them and the graphical representation is very intuitive.

6. Application/The use of data mining

Deployment

- Discuss how the result of the data mining will be deployed.
- Discuss any issues the firm should be aware of regarding deployment.
- Are there important ethical considerations?
- Identify the risks associated with your proposed plan and how you would mitigate them.

After evaluation, the Random Forest model with 50 embedded trees and maximum depth being 3 should be adopted in the future application. Companies can directly use this model for their application. This result can help institutions like banks to predict whether customers will violate the clauses of repaying credit cards on time, or the probabilities of their default so that banks can take actions as soon as possible to cope with such potential 'bad' situations. For the customers that are labeled with risk, companies can send messages to the customers or freezing the accounts.

However, there are still something that company need to pay attention to:
False negative instances. After observing the confusion matrix, the model tends to predict many false negatives. False negatives is fraud instances but being labeled with non-fraud. This will result in the additional cost of the company. Hence domain-knowledge validation is needed in the final stage.

False positive instances. False positive instances meaning a customer will not fraud but being labeled as fraud customers. If company freeze their card or send alert messages to them, company will lose customers and have a corresponding cost as well.

Since the profit of keeping one customer and the cost of a customer fraud varies from situation to situation, and also different for different companies, we do not have valid data for the cost matrix and can not plot the profit curve for models in advance. However, we think that it could be a good approach to help the company to target possible fraud customers and we suggest that company make their own profit curve based on individual situation.

7. Reference

Flood, Morten Hansen. "Early Identification of High-Risk Credit Card Customers Based on Behavioral Data." *Norwegian Institute of Science and Technology*, 2017.

\bibitem{ref_url1}

Lukas Frei (2019, Jan 15). Detecting Credit Card Fraud Using Machine Learning, \url{https://towardsdatascience.com/detecting-credit-card-fraud-using-machine-learning-a3d83423d3b8}

\bibitem{ref_paper1}

Sagadevan, Saravanan & Malim, Nurul & Shu Yee, Ong. (2018). Credit Card Fraud Detection Using Machine Learning As Data Mining Technique.

\bibitem{ref_paper2}

Minastireanu, Elena-Adriana, and Gabriela Mesnita. "An Analysis of the Most Used Machine Learning Algorithms for Online Fraud Detection." Informatica Economica 23.1 (2019).

\bibitem{ref_paper3}

Banerjee, Rishi, et al. "Comparative Analysis of Machine Learning Algorithms through Credit Card Fraud Detection." (2018).