Home Credit Default Risk Prediction

– Cai Xiwen, Sheng Xiangyu, Wang Hairuo, Zhu Yilin

Abstract

Concerns over risk management in banks have risen sharply ever since the Subprime Mortgage Crisis in 2008, and credit risk is one of the single largest risks faced by commercial banks among all the financial risks. And artificial intelligence technologies and updated algorithms are applied more in financial applications, particularly in the risk prediction problem. However, in terms of building predictive models on highly imbalanced sample, it is a quite challenging work for machine learning algorithms. And recent studies mostly focus on enhancing the classifier performance for credit default prediction. Consequently, in this study, various data resampling techniques are employed to overcome the issue of the data imbalance. Also different machine learning models are also employed to obtain efficient results. The results obtained are thoroughly evaluated from different angles based on many performance metrics, so the capacities of predictive models built by different combinations of resampling methods and machine learning algorithms can be appropriately evaluated. As a result, relatively better models can provide some valuable insights that can explain customer profile and behavior, which will help commercial banks, financial organizations, loan institutes, and other decision-makers to predict financial default risk earlier and prevent the substantial losses.

Keywords :

Imbalanced sample, Credit risk, Machine learning, Resampling techniques, Neural network, Risk prediction.

1. Introduction

Since the Subprime Mortgage Crisis in 2008, risk management in commercial banks has gained more attention. Meanwhile, many researches about artificial intelligence technologies and machine learning algorithms in business applications are booming both in academia and industry. It is expected that machine learning will be implemented more across multiple areas in a bank's risk management. For example, credit risk, also known as loan default risk, is one of the significant financial challenges in banking and financial institutions since it involves the uncertainty of the borrowers' ability to perform their contractual obligation, banks and financial institutions can build more accurate risk prediction models by identifying complex, nonlinear patterns within large datasets with machine learning algorithms and be able to identify the most relative characteristics that are indicative of people who have a higher probability of default on credit.

In this regard, our study proposes an analysis of a dataset of customers with over 200 features where a part of them were unable to repay their loans and got into default status. By using the methodology of data mining and machine learning algorithms, a series of predictive models were developed using classifiers such as, Gradient Boosting, SVM, XGBoost, Logistic Regression and Random Forest in order to evaluate the probability of a customer entering loan default or not. However, The performance of classifiers is compromised when the sample is highly imbalanced classifiers focused on the prior information that the majority class is way more than the minority class. So some additional methods would be conducted to get a better modeling result. Our goal is to provide some valuable insights that can predict potential defaults, even turn the study into some practical applications to prevent the financial default risk.

In the following sections, section 2 provides an overview of literature reviews on this topic. Section 3 gives a quick introduction to the data we use and the process of data cleaning. Section 4 begins by providing an overview of the research methodology including 7 resampling methods and 6 machine learning algorithms. Section 5 discusses the results and compares the performance of different models from the study. Section 6 focuses on conclusion and future work.

2. Literature review

2.1 Statistical and Machine Learning methods for Credit Risk Prediction

The assessment of Credit Default Risk is a popular topic in the banking and finance field. The accuracy of the Credit Risk Prediction of personal or business failure plays an important role in financial institutions. To increase the accuracy of the prediction, more and more Statistical and Machine Learning Methods were proposed for Credit Risk Prediction.

Logistic Regression (LR) is a traditional statistical method, the logic behind Logit model is to find an optimal linear relationship between dependent variables and explanatory variables (Wright 1995). In the credit risk industry, logistic regression remains the benchmark because it has simple interpretability and satisfies the requirements of financial regulators (Chen et al. 2016). Recent studies show that Machine Learning and Intelligence methods are more effective, and over perform traditional methods. The machine learning Method used in credit risk prediction recently includes Decision Trees(DTs), Random Forests, Artificial Neural Networks(ANNs), Support Vector Machines(SVMs), Extreme Gradient Tree Boosting (XGBoost) and the Semi-parametric method among others. In recent reviews, Tree-related methods such as decision trees, random forest, Classification and regression trees (CART) have also been widely used in the credit risk area and gradually become one of the standard models for credit risk prediction. In this paper, Lessmann et al. (2015) performed 41 classification algorithms across 8 credit scoring datasets and used different accuracy indicators to evaluate. The results of their research shows that random forests, which is the randomized version of decision trees, provide the best accuracy result. Another proposal for credit risk assessment is the semi-parametric method which combines the traditional and advanced machine learning methods. Li et al.(2014) combined a logistic regression model and some non-parametric methods such as SVMs and DTs and the result shows that the combination model can significantly improve the overall performance on default risk prediction. In recent years, many papers report good accuracy results for credit default prediction with the use of Artificial Neural Networks (ANN). For example, Mohammadi and Zangeneh (2016) designed Multilayer Perceptron Neural Network Models (MLP) trained using 6 different Back-Propagation (BP) algorithms, and chose the best performance model to compare with other classification algorithms such as LRs and DTs. Then they confirmed that the Artificial Neural Networks techniques perform better than other machine learning algorithms. In recent work, Extreme Gradient Tree Boosting (XGBoost) which could handle missing values with no need for imputation, became a hot topic for the research of credit risk prediction. In the credit risk industry, the imbalanced class is the main problem when using machine learning algorithms to predict the client's risk of default. In this study, Marceau et al.(2019) applied XGBoost and other machine learning methods to the imbalanced data, and the results showed that XGBoost overperforms other simple machine learning techniques.

2.2 Imbalanced data solution

For machine learning applications in the credit risk area, the imbalanced dataset is a big challenging task to train models and provide accurate performance. In recent studies, there are three methods to handle the imbalance data issues which are data-level, algorithm-level and hybrid methods (Branco et al. 2016). In terms of the data-level method, the re-sampling method which includes oversampling and undersampling methods are widely applied to the credit imbalance data. Alam et al. (2020) applied 3 undersampling and 6 oversampling techniques to the data and evaluated the performance to determine which resampling techniques provide good performance. The result shows that the oversampling techniques perform better than undersampling techniques for credit risk prediction. And for those three credit datasets in this paper, applying Gradient Boosted combined with K-means SMOTE provides the most accurate result. In recent work, many advanced methods are proposed to balance the dataset. For instance, Zhao et al. (2020) created the WHMBoost method which combines two resampling techniques and two base classifiers and both of them could be assigned weights to classify the imbalanced data well.

3.Data

3.1 Data Information

The dataset related to this project was provided by Home Credit Default Risk Kaggle Competition. There are 203 features and more than 300000 observations in this dataset. Each observation corresponds to a customer account which includes the client's information, and this information is organized by 203 features. These 203 features are related to the financial status of the clients which could be used to analyze the probability of default. This dataset also contains a binary target variable (TARGET) that represents the credit statutes, value of one means the client with repayment difficulties and the value of zero represents the client repaying loans on time. This dataset has a highly imbalanced problem, there are 226038 cases representing clients that have the good financial health to repay on time and 19970 cases belonging to client default.

Table 1: The description of the Top 5 positive and negative correlations features of the dataset

<table>
<tr><td><b>Target Values</b></td><td><b>Credit Risk (binary)</b></td><td>1 - client with payment difficulties
0 - client repay loans on time</td></tr>
<tr><td colspan="3"><b>Top 5 Positive Correlations</b></td></tr>
<tr><td><b>Feature Name</b></td><td colspan="2"><b>Feature Description</b></td></tr>
<tr><td>REGION_RATING_CLIENT_W_CITY</td><td colspan="2">Rating of the region where clients live taking the city into account</td></tr>
<tr><td>REGION_RATING_CLIENT</td><td colspan="2">Rating of the region where clients live</td></tr>
<tr><td>INCOME_WORKING</td><td colspan="2">The source of income is working</td></tr>
<tr><td>CODE_GENDER</td><td colspan="2">Gender of the client</td></tr>
<tr><td>DAYS_ID_PUBLISH</td><td colspan="2">How many days before the application did client change the identity document with which he applied for the loan</td></tr>
<tr><td colspan="3"><b>Top 5 Negative Correlations</b></td></tr>
<tr><td><b>Feature Name</b></td><td colspan="2"><b>Feature Description</b></td></tr>
<tr><td>DAYS_BIRTH</td><td colspan="2">Client's age in days at the time of application</td></tr>
<tr><td>EDUCATION_HIGHER</td><td colspan="2">Level of highest education the client achieved</td></tr>
<tr><td>DAYS_LAST_PHONE_CHANGE</td><td colspan="2">How many days before application did client change phone</td></tr>
<tr><td>DAYS_EMPLOYED</td><td colspan="2">How many days before the application the person started current employment</td></tr>
<tr><td>INCOME_PENSIONER</td><td colspan="2">The source of income is pensioner</td></tr>
</table>

3.2 Data Processing

1) Missing value processing

After importing the data, it could be found that some features of this dataset are incomplete with missing values. In this case, we filled in the missing values instead of dropping that feature because we think the observation with missing value could also be informative. In terms of the categorical variable, we use the most frequent values of that feature to fill in missing values. For the numerical variables, we tend to fill missing values with the median of the variables which is a more stable method to reduce the effect of the outlier.

2) Data Normalization & Data Split

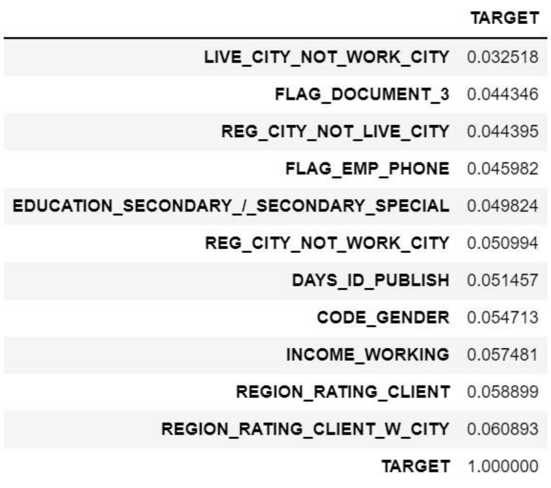
Normalization is changing the numeric value in the dataset to use a similar scale such as [0,1]. In machine learning, data normalization is important which could provide data consistency within the dataset and improve the training stability of the model (Ali et al. 2014). In this project, we applied the StandardScaler function to normalize the dataset.

After data cleaning and data normalization, we split the dataset into a training and testing set, considering 80% of the data for the model fitting and 20% for test purposes.

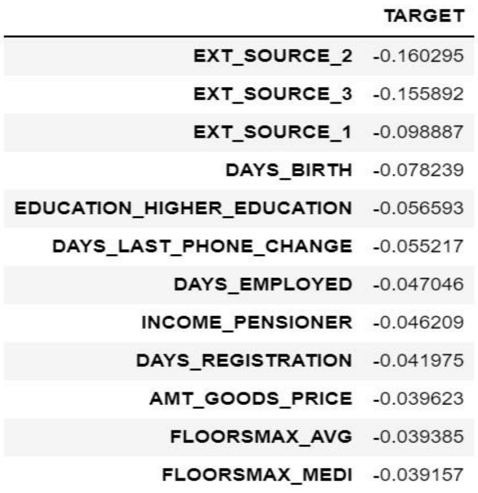
3.3 Data Visualization

Data Visualization is a powerful tool that enables financial institutions to understand data well. Recently, more and more banks and financial institutions used visualization tools to analyze and glean insight from data. In terms of machine learning for credit risk, data visualization could determine the important features which have a significant influence on the target variable and also provide insight for feature engineering to create some useful features to predict the risk of clients' default.

The Top 12 positive and negative correlations between the features and the target variable are listed below. From these two figures, it could be found that the correlation coefficients are very small however it still could provide some ideas about feature engineering. Another method to understand the data is the feature importance of random forests which are listed in figure 3. From the feature importance, we could understand what features the machine learning algorithms take into account when doing the prediction.



	TARGET
LIVE_CITY_NOT_WORK_CITY	0.032518
FLAG_DOCUMENT_3	0.044346
REG_CITY_NOT_LIVE_CITY	0.044395
FLAG_EMP_PHONE	0.045982
EDUCATION_SECONDARY_/_SECONDARY_SPECIAL	0.049824
REG_CITY_NOT_WORK_CITY	0.050994
DAYS_ID_PUBLISH	0.051457
CODE_GENDER	0.054713
INCOME_WORKING	0.057481
REGION_RATING_CLIENT	0.058899
REGION_RATING_CLIENT_W_CITY	0.060893
TARGET	1.000000



	TARGET
EXT_SOURCE_2	-0.160295
EXT_SOURCE_3	-0.155892
EXT_SOURCE_1	-0.098887
DAYS_BIRTH	-0.078239
EDUCATION_HIGHER_EDUCATION	-0.056593
DAYS_LAST_PHONE_CHANGE	-0.055217
DAYS_EMPLOYED	-0.047046
INCOME_PENSIONER	-0.046209
DAYS_REGISTRATION	-0.041975
AMT_GOODS_PRICE	-0.039623
FLOORSMAX_AVG	-0.039385
FLOORSMAX_MEDI	-0.039157

Figure 1 : Top 12 positive correlations

Figure 2 : Top 12 negative correlations

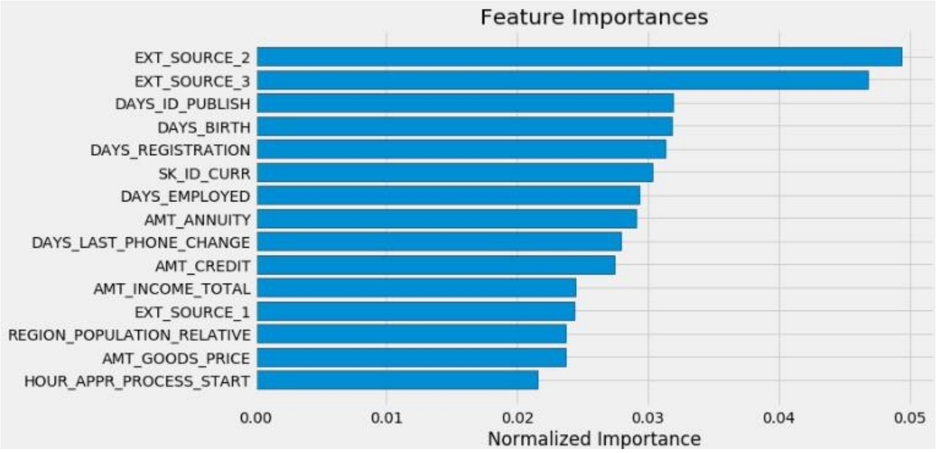
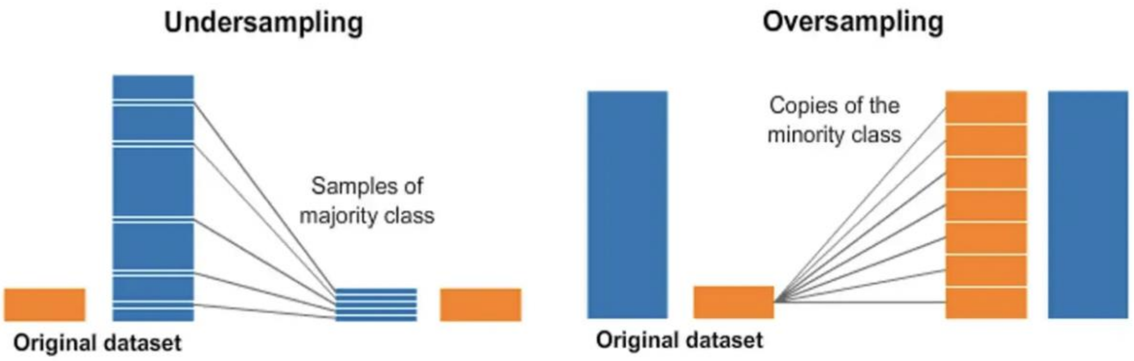


Figure 3: Important features from random forest

4. Methodology

4.1 Resampling Methods

Most of the machine learning algorithms perform better when the sample is almost balanced. But problems appear when a given sample like credit default is very imbalanced in nature. Classification of these imbalanced datasets is a very crucial task for the classifier as the classifier may tend to favor the majority class samples. As a result of unequal distribution of data, the majority class significantly dominates the minority class. To deal with such imbalanced learning problems many oversampling as well as undersampling techniques are available. Oversampling is an intuitive method that increases the size of the minority class, on the other hand, undersampling is to use a subset of the majority class to train the classifier. In real world applications, both methods have showed positive effects on the performance of many classifiers



Oversampling Methods

1) RandomOverSampling

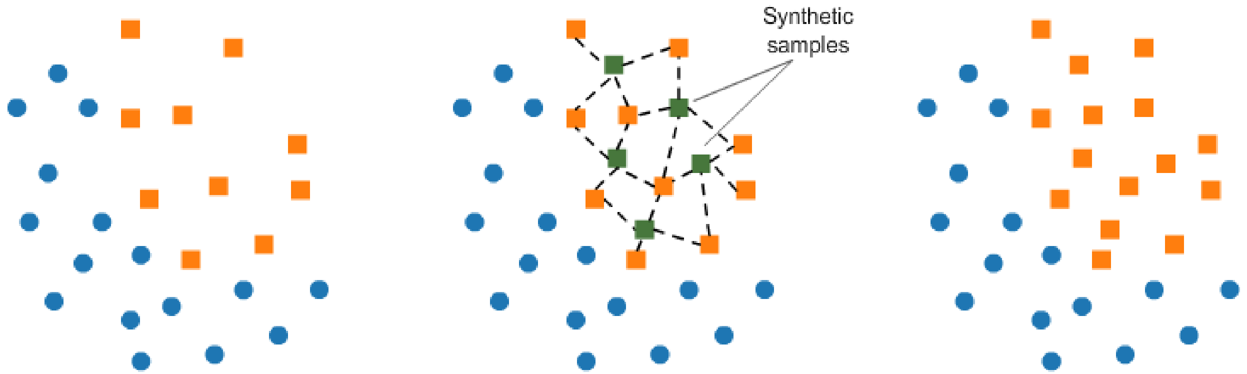
Random oversampling duplicates examples from the minority class in the training dataset and it might result in overfitting for some models because the replacement process is totally random and it only creates existing examples in the original minority class. It is the simplest strategy to do oversampling and is the fundamental concept of oversampling technique. Many other common oversampling algorithms used in real-world applications are developed based on this method.

2) Adaptive Synthetic Sampling Approach (ADASYN)

The essential idea of ADASYN is to use a weighted distribution for different minority class examples according to their level of difficulty in learning, where more synthetic data is generated for minority class examples that are harder to learn compared to those minority examples that are easier to learn. As a result, the ADASYN approach improves learning with respect to the data distributions in two ways: (1) reducing the bias introduced by the class imbalance, and (2) adaptively shifting the classification decision boundary toward the difficult examples. Simulation analyses on several machine learning data sets show the effectiveness of this method across five evaluation metrics.

3) Synthetic Minority Oversampling Technique (SMOTE)

SMOTE is known as the most common oversampling technique. Unlike random oversampling, SMOTE is an algorithm that performs data augmentation by creating synthetic data points based on the original data points. So it can be seen as an advanced version of oversampling. The SMOTE algorithm creates synthetic examples based on the feature space, rather than data space, similarities between existing minority examples, which is shown below.



4) Borderline SMOTE

In this algorithms, only the minority examples close to the **borderline** are being over-sampled, which adaptively creates a clearer separation of two classes. Borderline oversampling has been used widely for imbalanced data classification. It also classifies any minority observation as a noise point if all the neighbors are the majority class and such an observation is ignored while creating synthetic data.

Undersampling Methods

1) RandomUnderSampling

Random undersampling is a naive technique that involves randomly selecting examples from the majority class and deleting them from the training dataset. Because in the random undersampling, the majority class instances are discarded at random until a more balanced distribution is reached, it might result in losing information invaluable to the model.

2) NearMiss Algorithm

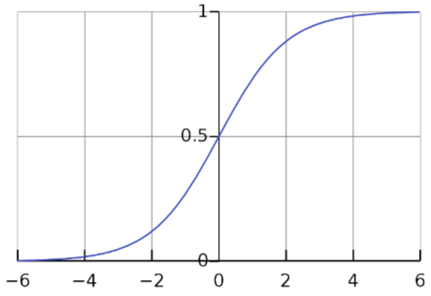
NearMiss is an under-sampling technique. This algorithm first calculates the distance between all the points in the larger class with the points in the smaller class. This can make the process of undersampling easier. Select instances of the majority class that have the shortest distance with the smaller class. These n classes will be stored for elimination. If there are m instances of the smaller class then the algorithm will return m*n instances of the majority class to be removed. NearMiss methods are widely used to prevent problem of information loss in most undersampling techniques.

3) NeighbourhoodCleaningRule

The Neighborhood Cleaning Rule is an undersampling technique that combines both the Condensed Nearest Neighbor (CNN) Rule to remove redundant examples and the Edited Nearest Neighbors (ENN) Rule to remove noisy or ambiguous examples. This approach involves first selecting all examples from the minority class. Then all of the ambiguous examples in the majority class are identified using the ENN rule and removed. Finally, a one-step version of CNN is used where those remaining examples in the majority class that are misclassified against the store are removed, but only if the number of examples in the majority class is larger than half the size of the minority class.

4.2 Data Modeling (6 algorithms)

1) Logistic Regression

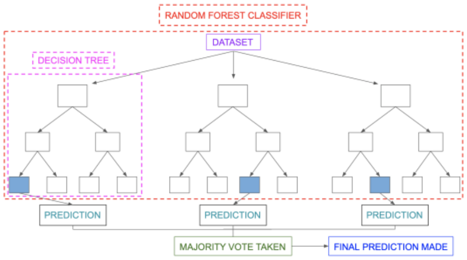


Advantages and disadvantages: Logistic regression is much easier to set up and train than other algorithms, which means it is much faster. Another advantage is that when the data is linearly separable, it is one of the most efficient algorithms. However, logistic regression tends to underperform when there are multiple or non-linear decision boundaries.

Brief introduction: Through estimating the parameters of a logistic model, logistic regression could be used to predict the binary variable and distinguish the binary class of the target variable.

Performance: Since the logistic regression is a kind of basic algorithm, at the beginning we didn't expect it could give us a good result. There is a parameter, ‘max_iter’(int, default = 100), we can set it as 1000, to increase the maximum number of iterations taken for the logistic regression model to improve the accuracy of the classifier. It is not outstanding on the original data, however, after resampling, the effect of logistic regression is surprisingly good.

2) Random Forest

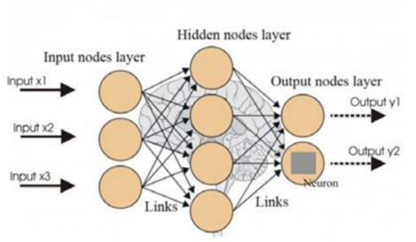


Advantages and disadvantages: Random forest can handle large datasets efficiently. The random forest algorithm provides a higher level of accuracy in predicting outcomes over the decision tree algorithm. However, it also requires much time for training as it combines a lot of decision trees to determine the class.

Brief introduction: The random forest algorithm is an extension of the bagging method as it utilizes both bagging and feature randomness to create an uncorrelated forest of decision trees. Feature randomness, also known as feature bagging , generates a random subset of features, which ensures low correlation among decision trees (Breiman, 2001).

Performance: Because random forest can handle large datasets efficiently, we expect it can perform well, however, maybe because of the low correlation between features and target, the Random forest does not perform as we expected.

3) Artificial Neural Network



Advantages and disadvantages: Neural networks have the ability to work with incomplete knowledge, after ANN training, the data can produce output even with incomplete information. The loss of performance here depends on the importance of the missing information. Corruption of one or more cells of ANN does not prevent it from generating output. The disadvantages are hardware dependence and the duration of the network is long.

Brief introduction: Artificial neural network is a computing system inspired by the biological neural networks which constitute animal brains (Hardesty 2017). ANN began as an attempt to exploit the architecture of the human brain to perform tasks that conventional algorithms had little success with. They soon reoriented towards improving empirical results, mostly abandoning attempts to remain true to their biological precursors. Neurons are connected to each other in various patterns, to allow the output of some neurons to become the input of others (Wesley 1997). Each connection, like the synapse in the biological brain, can transmit signals to other neurons. Artificial neurons receive signals, process them, and send signals to the neurons connected to them. The "signal" at the connection is a real number, and the output of each neuron is calculated by the nonlinear function of the sum of its inputs. Neurons and edges usually have weights that adjust as learning proceeds

Performance: ANN shows very strong stability, it presents well in original condition, but it didn’t show better performance after resampling. In addition, it takes a long period of time to run this model.

4) Gradient Boosting

Advantages and disadvantages: Gradient boosting can be more accurate than random forests. Because we train them to correct each other's errors, they're capable of capturing complex patterns in the data. However, if the data are noisy, the boosted trees may overfit and start modeling the noise (Krauss et al., 2017).

Brief introduction: Gradient boosting is a machine learning technique used in regression and classification tasks It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees. When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms random forest. A gradient-boosted trees model is built in a stage-wise fashion as in other boosting methods (Pirvonesi and EL-Diraby, 2020), but it generalizes the other methods by allowing optimization of an arbitrary differentiable loss function (Hastie et al., 2009).

Performance: Gradient boosting needs long running time. However, it shows no better scores on AUC and recall than random forest. That is because it is based on decision trees, maybe other kinds of gradient boosting will perform better.

5) XGBoost

Advantages and disadvantages: XGBoost (Extreme Gradient Boosting) is a mature open source software library, we can use XGBoost to deal with imbalanced data and missing value without resampling or filling values.While the XGBoost model often achieves higher accuracy than a single decision tree, it sacrifices the intrinsic interpretability of decision trees (Sagi and Rokach, 2021).

Brief introduction: XGBoost is an implementation of gradient boosting machines created by Tianqi Chen, now with contributions from many developers.

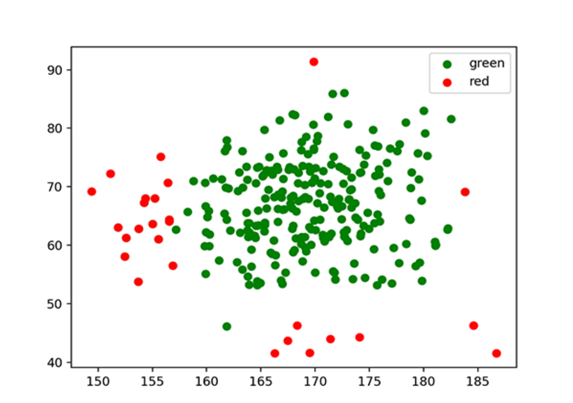
Performance: we can set parameter 'scale_pos_weight' as the quantity of result 0 divided by the quantity of result 1, in this way, imbalanced data can be effectively processed after running the algorithm. The result of XGBoost shows the best AUC under original data.

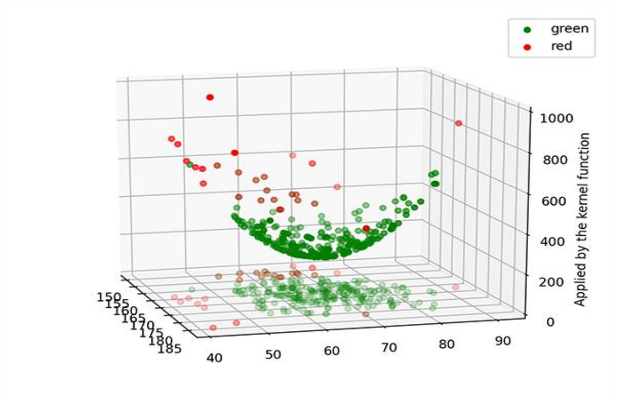
6) Support Vector Machines

Advantages and disadvantages: The advantages of support vector machines are that it is effective in high dimensional spaces and still effective in cases where the number of dimensions is greater than the number of samples. The disadvantage is when there are too many features, it will cause "Dimension explosion" when using SVM and lead to long time operation.

Brief introduction: support vector machines are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. SVM maps training examples to points in space so as to maximize the width of the gap between the two categories. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall (Corinna and Vladimir, 1995)

Performance: Our dataset has more than 200 features, so the SVM runs extremely slow. PCA can reduce the number of features down to 20, we are trying to perform SVM after PCA. Because logistic regression performs well in our project, we predict SVM will give us a good return.





4.3 Evaluation Metrics

Performance evaluation is the most important part in any predictive modeling task. It becomes even more critical in model predicting based on imbalanced sample, where the relative performance must be thoroughly evaluated from different angles. All the following 7 evaluation metrics will be conducted on various types of classifications.

1) Accuracy

$$Accuracy\ =\ \frac{TP+TN}{TP+TN+FP+FN}$$

Accuracy is a metric that generally describes how the model performs across all classes. It is useful when all classes are of equal importance. It is calculated as the ratio between the number of correct predictions to the total number of predictions.

2) Precision

$$Precision = \frac{TP}{TP+FP}$$

Precision refers to the number of true positives divided by the total number of positive predictions. The precision measures the model's accuracy in classifying a sample as positive.

3) Recall

$$Recall = \frac{TP}{TP+FN}$$

The recall is calculated as the ratio between the number of Positive samples correctly classified as Positive to the total number of Positive samples. The recall measures the model's ability to detect Positive samples. The higher the recall, the more positive samples detected.

4) F-1 score

$$F-1\ score = \frac{2*Precision*Recall}{Precision+Recall}$$

F1-score is one of the most important evaluation metrics in machine learning. It elegantly sums up the predictive performance of a model by combining two competing metrics — precision and recall

5) Specificity

$$Specificity = \frac{TN}{TN+FP}$$

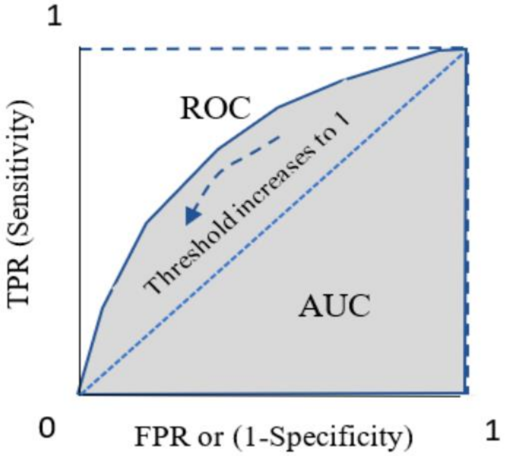
Specificity itself can be described as the model's ability to predict a true negative of each category available. It is also known simply as the true negative rate.

6) G-Mean

$$G-Mean = [(\frac{TP}{TP+FN})*(\frac{TN}{TN+FP})]^{\frac{1}{2}}$$

The Geometric Mean (G-Mean) is a metric that measures the balance between classification performances on both the majority and minority classes.This measure is essential for avoiding overfitting the negative class and underfitting the positive class

7) ROC/AOC Curve



A receiver operating characteristic (ROC) curve plot is also a widely used measure to evaluate the performance of classifiers. Specifically, the plot is created by plotting the true positive rate (recall) against the false positive rate at various threshold levels, which is shown above.

When comparing the performances of the predictive models, traditional metrics such as, accuracy, precision, and recall, do not give a meaningful evaluation, which is due to the nature of imbalance sample. For example, there is an imbalanced sample with 97% majority class and only 3% minority class. Just simply predict that all the results are majority class would give an accuracy of 97%. Therefore, a high overall predictive accuracy does not give much credit to the predictive model. Sometimes, recall is an important metrics for specifical situation, but a combination of these measures, such as G-mean used different combinations of specificity and sensitivity of the classifiers to give a better indication of performance; F-1 score is calculated with precision and recall. Also, the Receiver Operating Characteristic (ROC) curve is a good metric used in imbalanced situation to assess the performance of a classifier overall imbalance ratios and hence provide a summary of the entire range. As a result, in this study more focus would be on these three measurements, F-1 score, G-mean and ROC/AUC measure. The other evaluation metrics will be included in result, but mainly as a reference.

5. Result and Discussion

Table 1. Evaluation Metrics of Predictive Model with Initial Dataset.

Predictive Model	Precision (%)	Recall (%)	AUC	Accuracy(%)	Specificity (%)	G-mean (%)	F–1 Score (%)
Logistic Regression	84.84	92.11	0.5	92.1	100	92.11	88.32

Random Forest	90.38	92.12	0.5014	92.12	99.98	92.11	88.37
Gradient Boosting	88.64	92.1	0.5011	92.1	99.98	92.08	88.36
XGBoost	89.59	73.51	0.6814	73.51	74.52	54.78	79.33
Neural Network	89.06	76	0.6608	76	77.86	59.17	80.98

Table 1 shows the result of the evaluation metric of each predictive model based on the original dataset. We used the dataset which filled the missing value for all predictive models except the XGBoost because it could handle the dataset with missing values. We could see most evaluation metrics, including precision recall, accuracy, specificity, G-mean and F-1 score, have similar results for XGBoost and neural network are not high compared with the result of other predictive models. However, the AUC for XGBoost and neural network are 0.6814 and 0.6608 respectively, which represents the predictive models that could distinguish between repaying the loan class and non-repay the loan class in prediction. Moreover, XGBoost could handle the dataset without filling the missing value and not affect the prediction ability, which would reduce the time cost of data processing. As the time cost of processing data is higher and value of AUC is lower for neural network, then we would like to select the XGBoost as one of the final predictive models in probability of loan default prediction. On the other hand, the evaluation metric including precision, recall, accuracy, specificity, G-mean and F-1 score are high for the predictive models of logistic regression, random forest and gradient boosting. However, the AUC for logistic regression, random forest and gradient boosting are 0.5, 0.5014 and 0.5011 respectively. The result demonstrates that the classification predictive models do not have ability to distinguish between the repay the loan class and the non-repay the loan class for the dataset. Therefore, we could conclude that the logistic regression, random forest and gradient boosting do not have prediction ability based on the initial dataset because of the effect of imbalanced class problem for the target variable. The result of evaluation metrics indicates that we are supposed to perform resampling methods to the dataset to reduce the effect of imbalanced class problem of target variable for predictive models in prediction.

Table 2. Evaluation Metrics of Predictive Model with Dataset after Oversampling Methods

Over Sampling Methods	Predictive Models	Precision (%)	Recall (%)	AUC	Accuracy (%)	Specificity (%)	G-mean (%)	F-1 Score (%)
SMOTE	Logistic Regression	89.45	69.79	0.6795	69.79	70.14	48.96	76.58
	Random Forest	89.12	24.82	0.5519	24.82	18.97	4.7	30.45
	Gradient Boosting	88.86	21.61	0.5395	21.61	15.38	3.32	25.65
	Neural Network	89.11	62.23	0.6525	65.29	61.77	38.5	70.86
RandomOverSampling	Logistic Regression	88.66	68.98	0.6864	68.98	69.04	47.62	75.98

	Random Forest	87.61	91.54	0.5222	91.54	99.12	90.74	88.59
	Gradient Boosting	87.41	91.15	0.5307	91.15	98.49	89.78	88.63
	Neural Network	89.25	65.29	0.6639	65.29	65.07	42.48	73.18
Adaptive Synthetic	Logistic Regression	89.48	69.15	0.6788	69.15	69.37	47.97	76.1
	Random Forest	89.14	25.82	0.555	25.82	20.10	5.19	31.9
	Gradient Boosting	89.82	19.24	0.5379	19.24	12.59	2.42	21.76
	Neural Network	89.07	65.83	0.6581	65.83	65.83	43.34	73.6
Borderline - 1 SMOTE	Logistic Regression	89.38	71.96	0.6799	71.96	72.72	52.33	78.14
	Random Forest	89.03	31.52	0.5703	31.52	26	8.38	39.67
	Gradient Boosting	89.42	27.94	0.5653	27.94	22.42	6.26	34.31
	Neural Network	85.17	42.69	0.5008	42.69	41.26	17.61	53.51

Table 2 shows the evaluation metrics of each predictive model under the dataset with over sampling methods. There are four over sampling methods, which are SMOTE, Random Over Sampling, adaptive synthetic and Borderline - 1 SMOTE respectively. We could see that random forest and gradient boosting have obvious lower values of G-mean and F-measure across most over sampling methods, except the method of Random Over Sampling. However, the value of AUC for those two predictive models is still around 0.5, which means the over sampling methods do not reduce the effect of imbalanced class problem of target variable in prediction for random forest and gradient boosting. In the case of RandomOverSampling, we could see that the random forest and gradient boosting have elevated value of evaluation metrics but the value of AUC has not evidently increased compared with the result from the initial dataset. The result represents the RandomOverSampling based method that hardly reduces the effect of an imbalanced dataset to the prediction result for the predictive model of random forest and gradient boosting. On the other hand, we could see the logistic regression and neural network have better performance in prediction with the dataset after most oversampling methods. The value of AUC has obviously increased compared with the result from the initial dataset, and the value of G-mean and F-1 score for logistic regression and neural network are relatively high. As the result of evaluation metrics for predictive models are similar, that demonstrates the various over sampling methods have similar ability to handle the effect of the imbalanced data to the prediction result. Meanwhile, we could also see that logistic regression always has more elevated value of evaluation metrics under the dataset with oversampling methods compared with neural network. Thus, we could conclude that the logistic regression has the best performance with the dataset with oversampling methods in prediction. As most studies related to the imbalanced class problem, they would like to use the SMOTE to handle the imbalanced data. Therefore, we also would like to select the logistic regression and perform the SMOTE based method to handle the imbalanced data to predict the probability of loan default.

Table 3. Evaluation Metrics of Predictive Model with Dataset after Under Sampling Methods.

Over Sampling Methods	Predictive Models	Precision (%)	Recall (%)	AUC	Accuracy (%)	Specificity (%)	G-mean (%)	F-1 Score (%)
NearMiss	Logistic Regression	86.98	22.94	0.5223	22.94	17.29	3.97	28.09
	Random Forest	87.12	21.61	0.5219	21.61	15.72	3.4	26.01
	Gradient Boosting	87.2	21.74	0.5227	21.74	15.85	3.45	26.19
	Neural Network	88.15	15.97	0.5190	15.79	9.04	1.44	16.42
RandomUnderSampling	Logistic Regression	89.78	78.4	0.6866	68.78	68.81	47.33	75.84
	Random Forest	89.4	69.09	0.6763	69.09	69.37	47.93	76.06
	Gradient Boosting	89.34	70.26	0.6759	70.26	70.77	49.73	76.91
	Neural Network	89.46	64.39	0.6699	64.39	63.89	41.14	72.48
Neighbourhood CleaningRule	Logistic Regression	87.34	49.01	0.5691	49.01	47.49	23.28	59.43
	Random Forest	85.6	78.06	0.5192	78.06	83.1	64.87	81.45
	Gradient Boosting	87.74	91.88	0.5035	91.88	99.89	91.79	88.16
	Neural Network	89.38	69.43	0.6762	69.43	69.78	48.45	76.31

According to Table 3, which shows the evaluation metrics for each predictive model under the dataset with undersampling methods. There are three undersampling methods, which are NearMiss, RandomUnderSampling and NeighbourhoodCleaningRule. We could see the NearMiss and NeighbourhoodCleaningRule based methods do not noticeably increase the value of the evaluation matrix for the most predictive models, which represents those two under sampling methods do not have reduce the effect of imbalanced class problem of target variable within the dataset for the most predictive models. Except the predictive model of neural network with dataset with NeighbourhoodCleaningRule based methods. We could see the value of AUC for the neural network with dataset after performing the NeighbourhoodCleaningRule based method is 0.6762 and that is obviously increased compared with the result of the initial dataset. The result demonstrates the NeighbourhoodCleaningRule based method increased the ability for neural network to distinguish between repaying the loan class and non-repay the loan class. Moreover, the value of AUC for all predictive models with the dataset after resampling by RandomUnderSampling based method have significantly increased. The result demonstrates that the RandomUnderSampling based method could increase the ability for all predictive models to distinguish the class of repaying the loan and the class of not repaying the loan in this case. Furthermore, logistic regression has elevated value of precision, recall and AUC and the gradient boosting has elevated value of accuracy, specificity, G-mean and F-1 score with the dataset after resampling by RandomUnderSampling based method. As we would focus more on the AUC, G-mean and F-1 score to assess the prediction ability in this case because we applied the resampling method to the dataset. Thus, we could conclude that the gradient boosting has better prediction performance on this under sampling method according to the result of G-mean and F-1 score. As well, the value of AUC for gradient boosting is 0.6759, which is close to the result of AUC for logistic regression. Therefore, we have more evidence to select the gradient boosting with RandomUnderSampling based method as one of the final predictive models to predict the probability of default.

According to both Table 2 and Table 3, we could infer that all over sampling methods could reduce the effect of imbalanced class problems and improve the prediction ability for logistic regression and neural network in this case. Moreover, there is only the RandomUnderSampling based method that could improve the prediction ability for all predictive models within the under sampling methods in this case.

Table 4. Evaluation Matrix of Predictive Model with Dataset after PCA.

Predictive Model	Precision (%)	Recall (%)	AUC	Accuracy(%)	Specificity (%)	G-mean (%)	F–1 Score (%)
Logistic Regression	88.23	91.79	0.5204	91.79	99.46	91.29	88.65
Random Forest	88.48	91.91	0.5047	91.91	99.90	91.82	88.21
Gradient Boosting	92.73	92.11	0.5035	92.11	100	92.11	88.33
Neural Network	88.57	66.6	0.6304	66.6	67.27	44.81	74.27

Table 4 shows the result of the evaluation matrix of predictive models with the dataset after PCA. We could see that there is no significant change and increase for the evaluation matrix for each predictive model compared with the result from the initial dataset. Based on the result, we could conclude that the principal component analysis for the dataset in this case would not have a significant effect on the prediction result. Then we would not consider the predictive model with the dataset after principal component analysis as the final predictive model in predicting the probability of loan default.

Table 5. Predicted Probability of Personal Loan Default with each Predictive Model

Clients	XGBoost	Logistic Regression with SMOTE	Gradient Boosting with RandomUnderSampling
Client #1	0.8713	0.3714	0.2276
Client #2	0.2939	0.4456	0.5347
Client #3	0.3453	0.3302	0.3738
Client #4	0.1808	0.2179	0.5242
Client #5	0.8505	0.8699	0.7059

Table 5 shows the predicted probability of personal loan default with XGBoost, Logistic regression with SMOTE based method and gradient boosting with RandomUnderSampling based method. We could see there are some observations who have the different predicted probability of loan default. We inferred the differences among the probability of loan default because we used the different resampling method and we filled the missing value with the median of the numerical variable that would also provide misleading information to the prediction.

6. Conclusion and Future work

Overall, we are interested in making predictions about the probability of personal loan default and how to perform practical applications to make risk management for financial institutions. First of all, we found that the XGBoost could work with the dataset without filling the missing value and distinguish the class of repaying the loan and class of non-repaying the loan with relatively high accuracy. As well, it has a lower time cost of cleaning the dataset, then we would like to recommend using the predictive model of XGBoost to predict the probability of loan default. As well, we would conclude that the neural network also has great performance on predicting the probability of loan default based on the initial dataset. Moreover, the resampling methods that we applied do not have significant effect to improve the ability to distinguish the class of repaying the loan and the class of non-repay the loan for the neural network. For the over sampling methods that we use in the project, which has significantly improved the ability to distinguish the class of repaying the loan and the class of non-repay the loan for logistic regression and it does not work for the random forest and gradient boosting for this case. Furthermore, only the RandomUnderSampling based method has a significant effect on the prediction ability for predictive models within the three under sampling methods that we use in the project. Overall, the predictable probability of personal loan default exists the difference between observations for the different predictive models. We inferred that it is caused by the different resampling method and the way of filling the missing value. However, the financial institution also should consider the combinated result of predicted probability from different predictive models to make more rigorous decisions about providing the loan to clients or not.

As well, the prediction of probability of personal loan default could also be used to perform the risk management for financial institutions in practical application. The financial institution could collect the personal raw information from the client who wants to apply for the loan. Then, they could perform the data cleaning and processing for the dataset and put the features of clients as input for the predictive model to predict the probability of default of each client. Then the financial institution could use the predicted probability as credit rating score for each client and divide to provide the loan or not.

6.1 Limitation and Next Steps

As well, there are limitations for the predictive model and the predicted result mostly depends on the assumption when we applied the exploratory data analysis. Therefore, we found three primary problems and we want to solve them for the next step in this study. Firstly, we filled the missing value with the median of the numerical variable but that may not represent the commonest case for the variables. As well, we found that the different exploratory data analysis processes might have an impact on the performance of different predictive models. According to Qi et al. (2021) study, they filled the missing value with K nearest neighbours, which could ensure the filled missing value was more reliable. KNN could capture the pattern of the variable and provide more information with filling the missing values. Thus, we would also like to use the KNN method to fill the missing value of the numerical variables instead of median for the next time. We expect that method could provide more reliable information about the missing values.

As well, we would like to improve the machine learning algorithms that we use in the project because we only have four predictive models and we did not perform the hyperparameter tuning for the predictive models. For the next step, we would like to apply the hyperparameter tuning for the predictive models and attempt the various classification algorithms for the dataset to figure out the predictive model has better ability to distinguish the class of repay the loan and the class of non-repay the loan to increase the prediction accuracy in the next study with related topic of risk management.

Reference

Alam, T. M., Shaukat, K., Hameed, I. A., Luo, S., Sarwar, M. U., Shabbir, S., ... & Khushi, M. (2020). An investigation of credit card default prediction in the imbalanced datasets. *IEEE Access*, *8*, 201173-201198.

Ali, P. J. M., Faraj, R. H., Koya, E., Ali, P. J. M., & Faraj, R. H. (2014). Data normalization and standardization: a technical report. *Mach Learn Tech Rep*, *1*(1), 1-6.

Branco, P., Torgo, L., & Ribeiro, R. P. (2016). A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys (CSUR)*, *49*(2), 1-50.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, *20*(3), 273-297.

Chen, N., Ribeiro, B., & Chen, A. (2016). Financial credit risk assessment: a recent review. *Artificial Intelligence Review*, *45*(1), 1-23.

Hardesty, L. (2017). Explained: neural networks. *MIT News*, *14*.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Boosting and additive trees. In *The elements of statistical learning* (pp. 337-387). Springer, New York, NY.

Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, 259(2), 689-702.

Lessmann, S., Baesens, B., Seow, H. V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, *247*(1), 124-136.

Marceau, L., Qiu, L., Vandewiele, N., & Charton, E. (2019). A comparison of Deep Learning performances with other machine learning algorithms on credit scoring unbalanced data. *arXiv preprint arXiv:1907.12363*.

Mohammadi, N., & Zangeneh, M. (2016). Customer credit risk assessment using artificial neural networks. *IJ Information Technology and Computer Science*, *8*(3), 58-66.

Piryonesi, S. M., & El-Diraby, T. E. (2020). Data analytics in asset management: Cost-effective prediction of the pavement condition index. *Journal of Infrastructure Systems*, *26*(1), 04019036.

Qi, X., Guo, H., & Wang, W. (2021). A reliable KNN filling approach for incomplete interval-valued data. *Engineering Applications of Artificial Intelligence*, 100, 104175.

Sagi, O., & Rokach, L. (2021). Approximating XGBoost with an interpretable decision tree. *Information Sciences*, *572*, 522-542.

Wright, R. E. (1995). *Logistic regression*.