Traditional Machine Learning for Home Credit Default Risk

Kayode Awe, Maliha Zakir, Melissa Liao, Serhiy Chaykovskyy
Department of Electrical and Computer Engineering
University of Calgary
{kayode.awe, maliha.zakir, melissa.liaochen, serhiy.chaykovskyy}@ucalgary.ca

ABSTRACT

The most important questions in the banking and money lending industry are: How risky is the borrower and would they be able to repay the loan? And, should we lend them money?[12] Knowing the answers to these questions, financial institutions can determine if they will be able to gain interest and get back their investment or will the borrower default and they will lose on their investment. To answer these questions, lenders would often examine borrowers credit history, however for new clients without history it is difficult to determine credit risk. In this report we trained and compared several traditional machine learning classifier models that utilize multiple features non-related to credit history to help determine the probability of a borrower not defaulting on investment. The training and test dataset "puzzle" [13] was provided by Dr. Roberto Souza for the in-class example. Four different models were initially used to make predictions on the probabilities of defaulting. However, due to the imbalance nature of the database, four additional models were used in order to handle the imbalance nature of the data. The results of both sets of models were compared in terms of their accuracy metrics for confusion matrix. The result of analysis showed that EasyEnsembleClassifier has the best F1 score of 68.1%.

1. INTRODUCTION

Providing loans is one of the most important tools for financial institutions. In order to generate more profits financial institutions would need to provide more loans, however there is inherent risk in giving loans to borrowers who may not be able to fully repay the loan or completely default on it. For this reason, financial institutions need to examine each borrower's credit history in order to determine the risk involved with borrowing money to them. If the lender is too strict, it would eliminate a lot of potential borrowers and in turn would reduce the interest that they can generate from loans. Another problem is screening borrowers who do not have significant credit history, such as students or recently immigrated individuals. In order to assess their ability to repay the loans other than credit history, statistics would need to be included in the assessment. In this study we are planning to use traditional machine learning models to predict the possibility of each borrower to default on the loan.

2. RELATED WORK

Loan prediction is a highly sought-after subject in the sectors of banking and finance. Furthermore, following the tremendous improvements in data science and several remarkable developments in the field of artificial

intelligence, this topic has gained more attention and research interest. Over the last few years, researchers and the banking authorities have opted for training classifiers based on a multitude of machine learning and deep learning algorithms to automatically predict the credit score of an applicant based on their credit history and other historical data which made the process of selecting the eligible candidates a lot easier before the loan is approved. Due to the ever-increasing demands of loans now, the necessity for further improvements in the models for credit scoring and loan prediction is escalating significantly. Unlike previously, where experts were hired and the models depended on professional opinions were used for assessing the individual's creditworthiness, the focus has shifted to an automated way of doing the same task. Many noteworthy conclusions have been drawn in this regard which serve as stepping-stones for propitious research and studies.

For predicting loan default in the peer-to-peer(P2P) lending industry, the Random Forest Algorithm was adopted by Lin Zhu et al. [1]. The results were compared with the other three algorithms of logistic regression, decision tree and support vector machine. The experiment shows that the random forest algorithm performs outstandingly with an accuracy of 98%, higher than the other three algorithms in the prediction of loan default and has strong ability of generalization. Nonetheless, due to the importance of understanding and managing the risks in volatile business domains, it is crucial to find an effective aid in making decisions. The results of a comparison study on business analytics show that Random Forest Trees algorithm is a promising opportunity for predicting credit risk [2]. The main advantages of using Random Forest Trees in prediction are the competitive classification accuracy and simplicity. Such simplicity makes it easier for decision makers to understand the underlying relations, especially for the fact that none of the classification approaches achieved significant accuracy. In order to reduce the risk of mortgage loans, it requires a great attention to detail of each applicant's loan history and walking the fine line of who should and should not be approved. Credit applications that do not pass certain requirements are often not accepted due to the probability of them not paying back is high [3]. Meanwhile, low-income applicants are more likely to get approval, and they are more likely to pay back their loans in time. To find and examine the nature of the client applying for the personal loan, an exploratory data analysis technique was employed to gauge whether the customer can pay back the amount or not [4]. The result of such analysis shows that short term loans are preferred by the majority of the clients and the clients majorly apply for loans for debt consolidation. There are many instances where the data set is inconsistent due to missing values and anomalies. Hence it is essential to apply pre-processing algorithms. To deal with missing values, the K-NN algorithm was applied [5]. Additionally, the Binning algorithm was used for removal of these anomalies. These algorithms improved efficiency and made the data set more consistent.

3. MATERIALS AND METHODS

This section specifies the methodology and the materials used in the project. We are a team of four. Each team member worked on two models. This is to allow us work efficiently as some of the models took days to train due to the large amount of dataset involved.

3.1. Exploratory Data Analysis

The dataset used is from the Kaggle competition on Home Credit Default Risk. The target feature, which shows whether the loan was repaid or not, was inspected and analyzed. It was discovered that about 84% was categorized as 'False' while 16% was categorized as 'True'. This is shown in the diagram below.

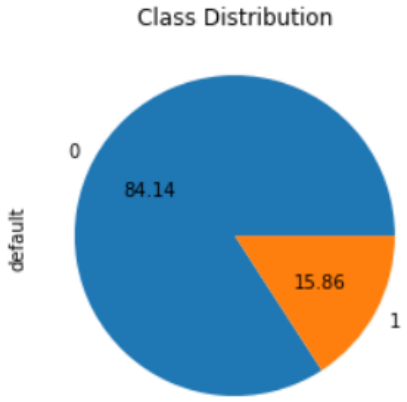

Fig. 1. Loan repayment analysis

The pie chart shows an unbalanced ratio which has a potential to cause low model accuracy. There were 64592 train samples, and 35000 test samples with 18 features left after dropping some features.

3.2. Data Preprocessing

One of the major steps employed in data processing for this project is to reduce or totally eliminate bias. As field of artificial intelligence is gaining prominence and playing crucial roles in businesses and society at large, there is need that models are designed in such a way as not to perpetuate discriminatory practices when being deployed [6]. To address this issue, we removed the features that we felt can lead to bias or discriminatory practices. In addition, we also removed features that won't contribute to the accuracy of the models.

On inspecting the training and the test datasets, a significant amount of missing data was found in some features. Binary features flags for missing values were created for all the features with missing values. Also, categorical features were represented as a series of binary values. Finally, categorical variables were encoded and NaN replaced by -1.

3.3. Model Building and Selection

The following models were used in order to compare their performances:

- LogisticRegression
- RandomForestClassifier
- GradientBoostingClassifier
- XGBoost

We tried to improve each model's generalization performance by tuning their parameters using Grid Search which tries all possible combinations of the parameters of interest in each model.

Scikit-learn GridSearchCV was used in conjunction with five-fold cross-validation to optimize the parameters of the models. We specified multiple potential values for the hyperparameters so that GridSearchCV can test all the possible combinations of parameters for us.

The following hyperparameters were used based on the GridSearchCV results:

Table 1
Optimal hyperparameters achieved for models

Model	Hyperparameters
LogisticRegression	C=0.1
	fit_intercept=True
	penalty=l1
	solver=liblinear
RandomForestClassifier	max_dept=50
	n_estimators=400
GradientBoostingClasifier	n_estimators=50
	learrning_rate=0.1
XGBoost	n_estimators=100
	learrning_rate=0.1

It must be noted that these are the hyperparameters that were used as a result of the GridSearchCV, some other parameters were included in the code but most of them did for output formatting and speed of execution.

3.4. Handling Imbalanced Dataset

As mentioned earlier, class imbalance is one of the vital issues that reduce the performance of the designed ML classifiers [7]. For the problem we are trying to solve, the class of default is far more than those not defaulting.

In view of this, we used four methods from the imblearn ensemble module, in addition to the previously mentioned classifiers. This was done in order to compare the performances of the methods from imbalance modules with the previous classifiers, especially to check for other metrics apart from accuracy score as this can be misleading when it comes to dealing with the problem of imbalance in a dataset.

The following methods were used for this purpose:

- EasyEnsembleCalssifier.
- RUSBoostClassifier.
- BalancedBaggingClassifier
- BalancedRandomForestClassifier

We used GridSearchCV to check for the optimum hyperparameters for the models.

3.5. Calculation of Metrics

We computed the Area Under Curve (AUC) values and the mean values for each of the models to aid in our comparison and analysis. We also computed the confusion matrix for each of the models.

To further gain more insight into the performances of the models, we computed the Accuracy, Precision, Recall and F1- Score for each model. We focused our attention on "recall" as it gives more objective evaluation of a model when it comes to imbalance problems [8].

Table 2
Basic evaluation measures from confusion matrix

Measure	Formula
ACC	(TP + TN) / (TP + TN + FN + FP)
ERR	(FP + FN) / (TP + TN + FN + FP)
SN, TPR, REC	TP / (TP + FN)
SP	TN / (TN + FP)
FPR	FP / (TN + FP)
PREC, PPV	TP / (TP + FP)
MCC	(TP * TN—FP * FN) / ((TP + FP)(TP + FN)(TN + FP)(TN + FN))$^{1/2}$
$F_{0.5}$	1.5 * PREC * REC / (0.25 * PREC + REC)
F_1	2 * PREC * REC / (PREC + REC)
F_2	5 * PREC * REC / (4 * PREC + REC)

3.6. Prediction

Since the target values for the test data were not available, we used the 'predict_proba' method from the estimators to infer the class probabilities, which is the probability that a particular data point falls into the underlying classes.

The prediction was carried out on the set of models that did not consider the effect of imbalance data and the other set of models from the imbalance-learn library.

4. RESULTS

Just like any real-world problem and in machine learning practice, the target labels are known in the training set but not in the test set. And based on the given datasets provided by the professor for this chosen project, the test dataset contains the same columns as the training set except for the 'Default' variable which is the target label column for our classification problem. Since we are not able to evaluate the accuracy of our prediction against the test set, other than only predicting the probability of default risk, the score results that are covered in this project are focused on the outcome from a stratified 5-fold cross-validation approach in our training set.

As mentioned in our materials and methods section, we have managed to train with 8 different learning methods for classification of default risk. Out of all 8 classifier models, 4 of them are typically used in traditional machine learning for balanced class distribution in dataset: Logistic Regression, Random Forest Classifier, Gradient Boosting Classifier and XGBoost. The other half of the list of models are specially used when dealing with imbalanced classes: Balanced Bagging Classifier, Balanced Random Forest Classifier, RUS Boost Classifier and Easy Ensemble Classifier. The results from these algorithms varied quite significantly across its scoring metrics but it did not differ much in the Area Under the Curve (AUC) score which yielded a range between 74% to 77%, given a difference of only ~3%.

On the other hand, the main differences on the rest of the scores was found to be driven by the type of model that it is intended for. With the implementation of two different sets of model type (models for balanced classes and models for imbalanced classes ratio), we observed that the first set of models (for balanced classes) performs much better in terms of accuracy and precision (specificity) score than the second set of models (for imbalanced classes), given approximately ~15% more accuracy and more than 30% in precision. For the recall (sensitivity) and F1-score, it is shown the opposite. Both scores produced higher percentages for the second set of models for imbalanced classes than those for balanced classes, given a difference of more than 47% in sensitivity and roughly ~20% in the F1-score. Details of the performance scores of each model are shown in Table 3.

5. DISCUSSION

Sometimes, the accuracy metric is not the ideal metric to use when we are evaluating an imbalanced dataset as it could be misleading. In a dataset with highly unbalanced classes, the classifier will always "predicts" the most common class without performing any analysis of the features and it will have a high accuracy rate, obviously not the correct one. [9] We have to understand the meaning behind the project in order to determine the optimal metric for the purpose of the problem or objective we are trying to achieve. In our case, the accuracy metric is not a good performance tool to go for this project as we are encountering an imbalanced classes ratio in our dataset. To mitigate this issue, it is best to look at minimizing the incorrect predictions given as positive class for those customers that would likely not pay back a loan i.e., False Negative, and maximize the number of correct predictions for actual default risky customers i.e., True Positive. Therefore, the F1-score and the recall (sensitivity) score are the most appropriate metrics to look at for this unbalanced dataset project. The F1-score is a harmonic mean of

Table 3
Performance Evaluation of each Algorithm on Training dataset

Learning Method	AUC	Accuracy	Specificity	Sensitivity	F1-Score
EasyEnsembleClassifier	0.769	0.711	0.312	0.681	0.428
BalancedRandomForestClassifier	0.760	0.702	0.303	0.676	0.419
RUSBoostClassifier	0.749	0.733	0.318	0.594	0.414
BalancedBaggingClassifier	0.754	0.692	0.295	0.678	0.411
XGBoost	0.763	0.853	0.630	0.178	0.277
GradientBoostingClassifier	0.765	0.854	0.652	0.166	0.265
RandomForestClassifier	0.750	0.851	0.674	0.123	0.207
LogisticRegression	0.756	0.850	0.644	0.124	0.207

precision and recall. It keeps the balance between precision and recall and improves the score only if the classifier identifies more of a certain class correctly and as for this project, it is the default risk that falls in the minority class. [10]

We can clearly see from the results section in Table 3, the 4 models that pertains to the usage of balanced dataset have poorer recall and F1-score performances than those exclusively for imbalanced dataset. Although the top best performances for unbalanced classes aren't ideal due to its percentage being less than 70%, it is better than the opposite set of models. Out of all 8 models we trained for this project, the model that has the overall best performance is the EasyEnsembleClassifier model. It has the highest score for the sensitivity and F1-score, yielding a 68.1% and 42.8%, respectively.

With that being said, it is important to consider the ratio and distribution number of classes across a dataset in machine learning and the intention behind the problem we are trying to solve. Our study has several strengths:

- The advantage of applying the EasyEnsembleClassifier model is the simple setup required to train the data and is very straightforward to use.
- General training time for each model in respect to the size of the dataset was relatively fast to run and compute.
- Similar experimental setup was utilized across all 8 models.

This study has some limitations as well:

- The main limitation of this study is the unbalanced classes ratio in the dataset. There are insufficient samples to let models learn the appropriate features to assert the default risk on loanees.
- From the given dataset, some of the features are not easy to interpret. It contains encrypted information in regards to the customer's profile. Hence, the process of feature engineering could be extended for more analysis.
- Further models can be trained to enhance the sensitivity (recall) and F1-score performance from the train dataset to improve the prediction results. Such models could include: random-under sampling, random-over sampling, synthetic minority oversampling technique (SMOTE) and penalized-algorithms. [9]

In comparison to previous studies, the SMOTE method was adopted to cope with the problem of imbalance class in the dataset but instead it focused on the accuracy method to evaluate the best model for the same study, given a 98% of accuracy with RandomForestClassifier model. [10]

6. CONCLUSION

The purpose of this experiment was to evaluate an individual's ability to return the loan borrowed from a financial institute using the Machine Learning approach. Therefore, multiple renowned classifiers such as GradientBoosting, LogisticRegression, XGBoost and RandomForestClassifier were adopted to evaluate and finalize a model with maximum accuracy for predicting loan default in the Home Credit dataset. Despite the fact that all the classifiers have performed exceedingly well with accuracy scores ~85%, XGBoost has yet lived up to its dominating reputation as a winner (85.4% accuracy). Nonetheless, as the dataset is prominently imbalanced between non-default/default proportion, we have implemented EasyEnsembleClassifier, BalancedBaggingClassifier, RUSBoosterClassifier and BalancedRandomForestClassifier to deal with

such issue. As the ‘default’ class is indubitably the most vital class in the dataset, we mainly focused on reaching a higher recall score. Thus, considering all possible trade-offs, we conclude that EasyEnsembleClassifier has the best performance.

Our work can be extended to a higher level in the future yielding a better efficiency by testing more classifiers such as K-Nearest Neighbors, Naïve Bayes, Decision Tree or perhaps a hybrid of different balanced and base models to invent the state-of-the-art loan default prediction algorithm.

REFERENCES

[1] Zhu L, Qiu D, Ergu D, Ying C and Liu K 2019 “A study on predicting loan default based on the random forest algorithm”, The 7th Int. Conf. on Information Technol. and Quantitative Management

[2] Ghatasheh N 2014, “Business analytics using random forest trees for credit risk prediction: a comparison study”, Int. Journal of Advanced Science and Technology

[3] Madane N and Nanda S 2019 , ”Loan prediction analysis using decision tree Journal of The Gujarat Research Society”

[4] Jency X F, Sumathi V P and Sri J S 2018, “An exploratory data analysis for loan prediction based on the nature of the clients”, Int. Journal of Recent Technol. and Engineering

[5] Kacheria A, Shivakumar N, Sawkar S and Gupta A 2016, “Loan sanctioning prediction system”, Int. Journal of Soft Computing and Engineering

[6] Lee, M. S. A. (2019). Context-conscious fairness in using machine learning to make decisions. AI Matters, 5(2), 23-29.

[7] A. J. Mary and S. P. A. Claret, "Imbalanced Classification Problems: Systematic Study and Challenges in Healthcare Insurance Fraud Detection," 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021, pp. 1049-1055, doi: 10.1109/ICOEI51242.2021.9452828.

[8] Saito, Takaya, and Marc Rehmsmeier. “The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets.” *PloS One*, vol. 10, no. 3, 2015, p. e0118432.

[9] B. Kumar, “Imbalanced classification: Handling imbalanced data using Python,” *Analytics Vidhya*, 24-Jul-2020. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/07/10-techniques-to-deal-with-class-imbalance-in-machine-learning/. [Accessed: 03-Apr-2022].

[10] S. Mazumder, “What is imbalanced data: Techniques to handle imbalanced data,” *Analytics Vidhya*, 23-Jun-2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/06/5-techniques-to-handle-imbalanced-data-for-a-classification-problem/. [Accessed: 03-Apr-2022].

[11] Pirjatullah, D. Kartini, D. T. Nugrahadi, Muliadi and A. Farmadi, "Hyperparameter Tuning using GridsearchCV on The Comparison of The Activation Function of The ELM Method to The Classification of Pneumonia in Toddlers," 2021 4th International Conference of Computer and Informatics Engineering (IC2IE), 2021, pp. 390-395, doi: 10.1109/IC2IE53219.2021.9649207.

[12] I. Dabbura, “Predicting loan repayment,” Medium, 15-Dec-2019. [Online]. Available: https://towardsdatascience.com/predicting-loan-repayment-5df4e0023e92. [Accessed: 02-Apr-2022].

[13] R.Souza, Class Lecture, Topic: “Traditional machine learning example.”, ENEL 645, Schulich School Of Engineering, University Of Calgary, Calgary, Mar. 14, 2022

GitHub Link

https://github.com/kayawemo/Final-Project-Group-20.git