Predicting Credit Risk with Machine Learning Models

Owen Tan Rena Pei Qi Chong Ming Feng Chua Alissa Alissa

Abstract

Credit risk assessment is an important task that every financial institution does to minimize financial losses from defaults. As such, credit risk prediction has been a popular research topic in today's society. Traditionally, financial institutions use statistical methods to determine an individual's credit risk. However, various research papers have suggested that machine learning can be used to better predict an individual's credit risk. This course paper aims to apply and evaluate the use of some machine learning techniques in credit risk prediction.

1. Introduction

Credit risk prediction is an important part of a financial institution's decision-making, as it assesses the likelihood of a borrower defaulting on their payments. Traditionally, statistical methods (eg. Altman z-score model) are used to assess credit risk. However, there are many limitations in the accuracy of this prediction as the statistics are only based on a few variables and are unable to accommodate the variety of variables that may affect the credit risk of an individual. In recent year, many studies have shown that the machine learning is able improve credit risk prediction accuracy as machine learning models can use large amounts of data and complex models to uncover patterns and relationships that statistical methods miss or fail to take into account.

This report explores the application of machine learning in credit risk prediction, providing an overview of different machine learning models, data pre-processing techniques, and an evaluation of the different machine learning models used.

2. Related Work

There have been numerous studies and applications of machine learning in credit risk prediction. A wide variety of machine learning algorithms have been used for credit risk prediction, including logistic regression [1], decision trees [2], neural networks [1], and support vector machines [3].

To improve credit risk prediction accuracy, we can use a variety of methods such as feature selection and data pre-processing techniques [4] on the dataset, before fitting the model to the dataset. Clustering techniques can also be used to group similar borrowers and improve the accuracy of a credit risk model [5]. Moreover, some studies have explored using unconventional data sources such as social media data [6] to predict individual credit risk.

Overall, these studies demonstrate that there is a big potential for the use of machine learning in credit risk prediction and that it is important to consider various machine learning models and features to achieve accurate results.

3. Methodology

3.1. Dataset

This dataset contains information about individuals from Germany who took a loan and their respective credit risk (good or bad) prepared by Prof. Hofmann [7]. There are 10 features in the data: age, sex, job, housing, saving accounts, checking account, credit amount, duration, purpose. These features are used to determine whether the person has a good or bad credit risk in our machine learning model.

3.2. Exploratory Data Analysis

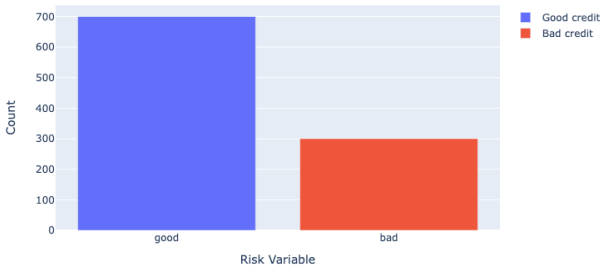

Figure 1. Distribution of good and bad loans taken

From Figure 1, we can see that the data set distribution leans towards good loans. However, in real life, there are more good loans than bad loans, thus this imbalance is to

be expected.

3.3. Data Pre-processing & Feature Engineering

We first preprocessed the data by handling missing values for columns ‘Saving accounts‘ and ‘Checking account‘. This is done by adding a new category ”no inf” to represent the missing information in these two columns.

We have also chosen to convert ‘Age‘ variable to a categorical variable. Ages 18 to 25, ages 26 to 35, ages 35 to 60, and ages above 60 was converted to ‘Student‘, ‘Young‘, ‘Adult‘ and ‘Senior‘ categories respectively.

Subsequently, one-hot encoding was applied to the 7 categorical variables ‘Age‘, ‘Saving accounts‘, ‘Checking account‘, ‘Purpose‘, ‘Sex‘, ‘Housing‘ and ‘Risk‘. Lastly, log transformation was applied on ‘Credit amount‘.

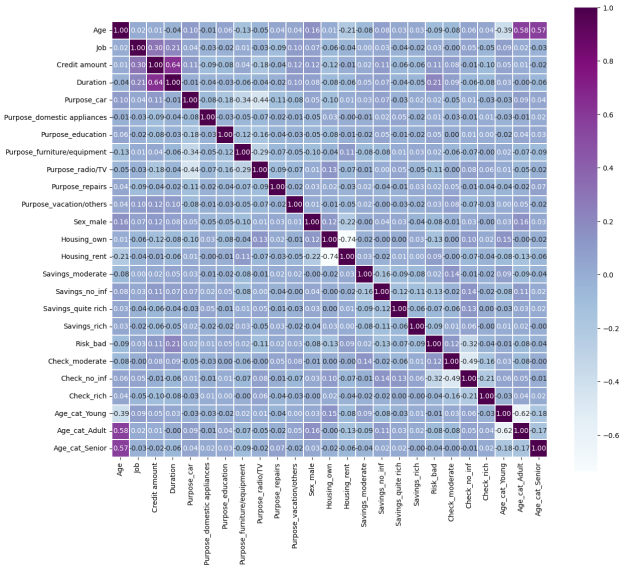

Figure 2. Correlation Heatmap

We also attempted to see if there are any redundant features in the dataset and plotted a correlation heatmap graph. From Figure 2, we can see that the highest correlation is between duration and credit amount (0.64). However, we decided to keep both features as the correlation was not strong enough to remove any of them.

3.4. Modelling

In this project, we applied Logistic Regression, Naive Bayes, Random Forest, and Decision Tree algorithms to the German Credit Risk dataset to predict credit risk. We first preprocessed the data by handling missing values, scaling numerical features, and encoding categorical features. We then split the data into training and testing sets, and trained each algorithm on the training data using cross-validation to tune hyperparameters and prevent overfitting. We evaluated the performance of each algorithm on the testing data using metrics such as accuracy, precision, recall, and F1 score. Our results showed that all four algorithms were able to achieve high accuracy in predicting credit risk, with Logistic Regression and Decision Tree performing slightly better than Random Forest and Naive Bayes. Overall, our study demonstrates the effectiveness of machine learning algorithms in credit risk analysis, and highlights the importance of feature selection and hyperparameter tuning in achieving optimal performance.

Table 1. Final hyperparameters used for decision tree model.

HYPERPARAMETERS	VALUE
MAX_DEPTH	10
MIN_SAMPLES_SPLIT	5
MIN_SAMPLES_LEAF	10
CRITERION	GINI

3.4.1. DECISION TREE

Introduction Decision tree is a supervised machine learning algorithm that can be used for classification. It is a tree-like model built using a series of feature-based splits to produce a prediction. Compared to other machine learning algorithms, decision trees are easy to interpret and implement. Furthermore, the decision tree model can be visualized, which makes them useful for understanding the factors that contribute to credit risk. However, decision trees are prone to overfitting, if the hyperparameters used to fit the model are selected incorrectly. In our study, we build a decision model and tuned its hyperparameters, and achieved an accuracy of over 70% in predicting credit risk.

Methodology We carried out hyperparameter tuning using GridSearchCV method to obtain the optimal hyperparameters for the Decision Tree model. The GridSearchCV method selects the best parameters for the model by performing cross-validation on models fitted on the different combinations of the hyperparameters. For our study, the parameters tuned included the maximum depth of the tree (max_depth), the minimum number of samples required to split an internal node (min_samples_split), and the minimum number of samples required to be at a leaf node (min_samples_leaf). The final hyperparameters can be seen from Table 1.

3.4.2. RANDOM FOREST

Introduction Random forest uses an ensemble of decision trees to make predictions. It consists of building multiple decision trees using samples of the dataset, and the output

Table 2. Final hyperparameters used for random forest model.

HYPERPARAMETERS	VALUE
N_ESTIMATORS	250
MAX_DEPTH	11
MIN_SAMPLES_SPLIT	6
MIN_SAMPLES_LEAF	1
CRITERION	GINI

of the model is given by the majority class predicted by the decision trees. Compared to decision trees, random forest can reduce the risk of overfitting by combining multiple decision trees. In our study, we build a random forest model and tuned its hyperparameters, and achieved an accuracy of over 75.6% in predicting credit risk.

Methodology In order to optimize the performance of the Random Forest model on the German Credit Risk dataset, hyperparameter tuning was carried out using the sklearn.model_selection::RandomizedSearchCV method. The hyperparameters that were tuned included the number of estimators (n_estimators), which represents the number of decision trees in the forest. Other important hyperparameters include max_depth, which controls the maximum depth of each decision tree, and min_samples_split and min_samples_leaf, which controls the minimum number of samples required to split an internal node and the minimum number of samples required to be a leaf node, respectively. By using the RandomizedSearchCV method, a range of hyperparameters was randomly sampled from the parameter space, and cross-validation was used to evaluate the performance of the resulting models. The best hyperparameters were then selected based on the mean cross-validation score. The final hyperparameters can be seen from Table 2.

3.4.3. LOGISTIC REGRESSION

Introduction Logistic regression is a model that can be used for binary classification by calculating the log odds of an event. It can be extended to handle non-linear relationships between features and the output variable through the use of polynomial terms, interaction terms, or other transformations. However, logistic regression is prone to overfitting, as fitting more variables to the logistic regression model will always increase the amount of variance explained in the log odds. As such, we can impose a penalty to the logistic regression model to prevent overfitting. In our study, we build a logistic regression model with penalization, and achieved good performance after tuning its hyperparameters.

Methodology The model's performance was optimized by tuning its hyperparameters. This is done through Grid Search Cross Validation. It is a method used to systematically search for the optimal hyperparameters of a machine learning model. It works by creating a grid of all possible combinations of hyperparameter values to test. There are a few hyperparameters that can be tuned for logistic regression to improve model performance. Using GridSearchCV, we tested with a combination of penalty, regularization parameter, solver and maximum iterations.

3.4.4. NAIVE BAYES

Introduction Naive Bayes can also be used to estimate the probability of an individual's credit risk given their features. In addition to its computational efficiency, naive Bayes is particularly useful when dealing with categorical or discrete features, which are common in credit risk analysis. Naive Bayes assumes that the features are conditionally independent given the class variable, which can be a limitation when there are strong correlations between features. However, in our study, we performed feature selection and engineering to reduce the impact of correlated features and ensure the validity of the naive Bayes assumptions. We then applied naive Bayes to the preprocessed dataset and achieved good performance in predicting credit risk. Our results demonstrate that naive Bayes can be a reliable and efficient tool for credit risk analysis, especially when the input features are categorical or discrete.

Methodology In Naive Bayes, the probability of a feature given a class is estimated as the relative frequency of that feature in the training data for that class. However, if a feature does not appear in the training data for a particular class, the relative frequency estimate will be zero. This can result in zero probabilities when calculating the class posterior probabilities using Bayes' theorem, which in turn can lead to poor classification performance. Smoothing is a technique used to avoid zero probabilities by adding a small constant value. For Naive Bayes, we can tune the amount of smoothing. A higher value of the smoothing hyperparameter corresponds to more smoothing, which can help to avoid overfitting and improve generalization performance.

4. Results and Discussion

Results From Table 3, our experiments revealed that Random Forest outperformed the other models, achieving the highest accuracy scores. For our Random Forest model, we also observe from Figure 3 that we obtained an AUC of 0.634, indicating a moderate ability to distinguish between good and bad credit risk applicants.

Discussion In terms of credit risk, Random Forest can perform better than decision trees for the following several reasons.

Since the credit risk dataset is of high complexity and of

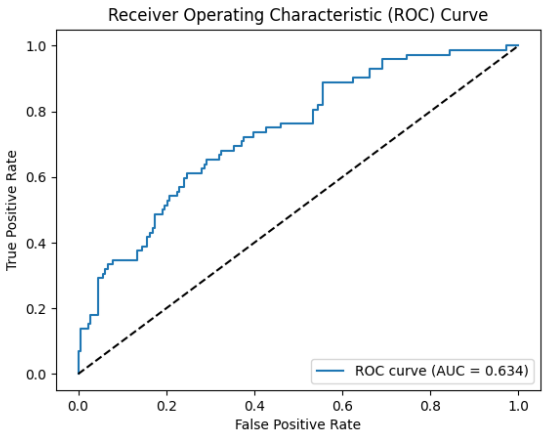

Figure 3. ROC curve showing the trade-off between true positive rate and false positive rate for the Random Forest Model

Table 3. Classification accuracies for various algorithms.

ALGORITHMS	ACCURACY
LOGISITIC REGRESSION	0.740
DECISION TREES	0.744
NAIVE BAYES	0.664
RANDOM FOREST	0.756

high dimensions, which can make it difficult to identify relevant features that accurately predict creditworthiness. Decision trees may not be able to capture the complex interactions between features in the dataset, leading to poor predictive performance. On the other hand, Random Forests can handle such complexity by using an ensemble of trees that take into account different combinations of features, making it more likely to capture the relevant features that are important for predicting creditworthiness.

The dataset could also contain noise and outliers that can negatively impact the performance of a single decision tree. Random Forests, on the other hand, are less sensitive to noise and outliers because they are built on multiple subsets of the data and features. The noise and outliers may affect some of the trees, but not necessarily all of them. Since the final prediction of a Random Forest is based on the majority vote of all the trees, it is less likely to be influenced by the noise and outliers.

Finally, credit risk prediction is a high-stakes task where the cost of making incorrect predictions can be significant. Random Forests are often preferred over decision trees because they tend to be more accurate and less prone to overfitting, which can reduce the risk of making incorrect predictions that can result in financial losses for lenders.

5. Conclusion

In conclusion, credit risk analysis is a crucial task in the financial industry that can be effectively addressed using machine learning techniques. In this article, we compared four popular machine learning models, namely Naive Bayes, Logistic Regression, Decision Trees, and Random Forest, on their performance in credit risk analysis.

Logistic Regression and Decision Trees produced the most accurate models. However, it is important to note that the choice of the best model ultimately depends on the specific needs and constraints of the business. Naive Bayes and Logistic Regression are simple and interpretable models that can be easily implemented while Random Forest is a more complex model that requires more computational resources but can deliver higher accuracy in some cases.

Overall, our comparison highlights the importance of selecting the right machine learning model based on the specific requirements of the credit risk analysis task. For future work, we can explore the use of ensemble learning techniques such as Random Forest to enhance the accuracy and robustness of credit risk analysis models based on Logistic Regression and Decision Trees.

6. Github

The link to our code can be found at: Final Project Github

We use the following dataset from Kaggle: German Credit Risk

Acknowledgements

We express our gratitude to Jorge Silva, who taught the Machine Learning course that inspired this paper. The course has provided us with a much-needed basic understanding of Machine Learning which will be useful in our future pursuits.

References

[1] Gouvêa, M. A., & Gonçalves, E. B. (2007, May). Credit risk analysis applying logistic regression, neural networks and genetic algorithms models. In POMS 18th annual conference.

[2] Satchidananda, S. S., & Simha, J. B. (2006). Comparing decision trees with logistic regression for credit risk analysis. International Institute of Information Technology, Bangalore, India.

[3] Lai, K. K., Yu, L., Zhou, L., & Wang, S. (2006). Credit risk evaluation with least square support vector machine. In Rough Sets and Knowledge Technology: First International Conference, RSKT 2006, Chongquing, China, July

24-26, 2006. Proceedings 1 (pp. 490-495). Springer Berlin Heidelberg.

[4] Ha, V. S., Lu, D. N., Choi, G. S., Nguyen, H. N., & Yoon, B. (2019, February). Improving credit risk prediction in online peer-to-peer (P2P) lending using feature selection with deep learning. In 2019 21st International Conference on Advanced Communication Technology (ICACT) (pp. 511-515). IEEE.

[5] Caruso, G., Gattone, S. A., Fortuna, F., & Di Battista, T. (2021). Cluster Analysis for mixed data: An application to credit risk evaluation. Socio-Economic Planning Sciences, 73, 100850.

[6] Yu, X., Yang, Q., Wang, R., Fang, R., & Deng, M. (2020). Data cleaning for personal credit scoring by utilizing social media data: An empirical study. IEEE Intelligent Systems, 35(2), 7-15.

[7] Accessed 2019. German credit dataset. https://archive.ics.uci.edu/ml/support/statlog+(german+credit+data).