DOI: 10.54254/2755-2721/74/20240495

Comparison of machine learning models in credit risk assessment

Yuchen Chen[1,6,†], Hongling Wang[2,*,†], Yuxuan Han[3,7,†], Yuxuan Feng[4,8,†], Hongchi Lu[5,9,†]

[1]Kingsway College Oshawa, Ontario, L1K 2H5 Canada
[2]Division of General Studies. University of Illinois Champaign, 61820, USA
[3]Computer science, University of Birmingham, Birmingham, B15 2TT UK
[4]Faculty of Science and Technology, University of Macau, Macau, China
[5]School of Natural Sciences, University of Manchester, Manchester, M13 9PL UK

[6]tommychen@kingsway.college
*Corresponding author: 350796581@qq.com
[7]yxh173@student.bham.ac.uk
[8]dc12795@um.edu.mo
[9]hongchi.lu@student.manchester.ac.uk
[†]All the authors contributed equally to this work and should be considered as co-first author.

Abstract. Credit risk plays an important role in finance. In a sense, credit risk reflects the stability of financial institutions and the protection of investors. Due to the impact of the epidemic on the world economy, we need to reassess the credit risk. For a long time, machine learning and deep learning in statistics have been very effective in predicting credit risk. But in machine learning and deep learning, when predicting credit risk, we cited four models, namely XGBoost, Decision Tree, Random Forest and Convolutional Neural Network (CNN) models. Analyze the advantages and disadvantages of these models: XGBoost can optimize the tree model and prevent overfitting through regularization, but XGBoost has poor interpretability. Decision Tree has strong explanatory power, but it is prone to overfitting. Compared with the decision tree, Random Forests increase accuracy and reduce the probability of overfitting, but Random Forests consume more time and computing resources. Although Convolutional Neural Network has a high accuracy rate, it abandons interpretability. Therefore, in our experiments, we found that these models are not perfect and have their own defects. So, in future research, we will make an integrated model to include the advantages of the model and discard some defects, so that the integrated model will have better generalization and accuracy.

Keywords: Machine Learning, Credit Risk, finance.

1. Introduction

1.1. Background

Credit risk is the risk of default on a debt that may arise due to the borrower's failure to make required payments, resulting in the possibility that the lender will lose its holding [1]. Due to the impact of COVID-19, the credit lending sector of the banking industry has seen a great recession in the lending market, leading to more uncertainties and financial risks for investors [2]. In response to the impact of the epidemic, bank loan growth has slowed down, and the adverse impact on bank loan growth largely depends on the severity of the epidemic in the country. Credit loans are loans issued by banks or financial institutions to borrowers that are repayable with or without an amount of interest. n order to reduce the risks of lending, banks, and financial institutions adjusted the requirements for assessing credit risks. This research focuses on the comparison between the use of the machine learning model and the deep learning model.

1.2. Problems

The pandemic poses the biggest threat to the economy like never before, which the economy of 2020 is considered to be worse than the economic depression in 2008-2009 [3]. The lending sector of the banking industry faces higher demands with risks of the borrowers not repaying back due to increase on the inflation rate, unemployments and costs of living. Hence, developing a precise and efficient credit risk model is crucial to reduce the risks of investments.

Traditional credit risk assessment models use credit scores as one of the most important metrics to determine credit risks. Credit scores are numbers used to depict a consumer's creditworthiness, and they are determined based on individual's payment history, amounts owed, credit length history, and other features which varies depending on the financial institution. The majority of banks or financial institutions recognize FICO score as the most used credit scoring system. However, credit scoring systems primarily rely on data from credit bureaus, which might not fully capture an individual's financial situation. And sudden financial circumstances might also impact an individual's willingness to repay, such as during the pandemic, more consumers demand higher loan defaults with lesser probability of repaying the banks. This will lead the credit risk assessment model to be less accurate.

1.3. Thesis Statement

We prepare to compare and analyze these four models for credit index. We are going to use each of the four models to train on the same dataset and compare their strengths and weaknesses to find the model that best matches the credit risk, and if the results are not satisfactory, we will choose to find an integrated learning model that combines the strengths of all the models to maximise the benefits of predicting the credit risk.

1.4. Structure

This paper is organized into 8 parts. Chapter 2 provides a succint literature reviewof relevant studies using machine learning methods to assess credit risk. Chapter 3 outlines the approaches that were used in the experiment, including Decision Tree, XGBoost, Random Forest, and CNN. In Chapter 4, exploratory analysis is conducted on the dataset we picked. Chapter 5 presents the evaluation metrics we used for the 4 models. Chapters 6 provides detailed information on the four models, and Chapter 7 listed the experimental results. Chapter 8 is a summary of the thesis and discusses possible future implemenations for research, and listed the experiments' limitations. Finally, Chapter 9 lists the references.

2. Literature Review

Using machine learning to predict credit risk has already been common among researchers worldwide. However, there are many models of machine learning.

Some studies use XGBoost to evaluate credit risk. In the experiment and pointed out that XGBoost can perform better when calculating credit risk. they standardized quantitative and qualitative indicators and set up an initial index system. Use Logistic regression, AIC-Logistic regression, and BIC-Logistic regression to filter features, and the best combination of indicators is determined by using various effective indicators such as AUC and accuracy. The variables are mainly about people's personal information. Finally, they verify the effectiveness of the XGBoost model [4].

As for the decision tree, Satchidananda and Simha compare the decision tree with logistic regression. They used 25 variables in the data, such as Crop for which the loan was taken, procured inputs, Spent for irrigation, etc. In the experiment, the K-means clustering algorithm is used to balance the positive and negative values from the negative samples, and the number of clusters is set to the number of positive samples to reduce the lack of information. The final experiment shows that the decision tree can perform better than the logistic regression [5].

The third experiment is about using random forests to evaluate credit risk. In the experiment, they used the machine learning public data of the University of California, Irvine, and divided the subjects into good credit and bad credit. 20 data such as the Status of an existing checking account, Duration in a month, and Credit history are used. All classification runs used 10-Fold cross-validation and used different methods such as HeuristicLab, Weka, and Keel. We benchmarked the random forest algorithm and other algorithms and found that the random forest has the highest Sensitivity, F-Measure, and Accuracy values. In the conclusion part, Ghatasheh [6] pointed out that random forest is a good method for testing credit risk because the classification accuracy is high and potential relationships can be found well.

The fourth Convolutional Neural Network (CNN) model integrates convolutional, pooling, and fully connected layers into a single neural network. Researchers generate feature maps using weighted matrix filters and then reduce them to smaller matrices through pooling to eliminate redundant data. In this process, they utilize three key characteristics of CNN: local connections, parameter sharing, and translational invariance. Meanwhile, they also emphasize the application of activation functions, linear mapping, and pooling functions. In this study, they use Hyperspectral Image Cubes (HSIC) to identify pixel vectors within HSI cubes. They view the classification problem as a mapping function, with the aim to minimize the difference between output values and predicted values. The research results show that the interdependence between risk variables is stronger under extreme market conditions. Traditional linear correlation coefficients cannot capture the nonlinear dependencies prevalent in financial markets. Therefore, the authors suggest modifying the optimization model to explicitly consider the objectives and constraints of passive investment. In conclusion, this research demonstrates the advantages of CNN in image recognition and feature extraction and its application in financial risk prediction model

3. Methodology

3.1. Decision Tree

Decision tree is a non-parametric supervised learning algorithm that uses a tree-like model of decisions and their possible consequences. Decision tree is typically generated in a form that is represented by a statistical classifier and can be used for clustering. There are nodes and branches in the decision tree. each node represents data and requires one or more properties, and each branch contains a set of classification rules, which can be found towards the end of the branch [7]. The decision tree is one of the most used machine learning algorithms, but it comes with advantages and drawbacks. The decision tree is easy to comprehend and requires few data preparation, and it can handle both numerical and categorical data. The decision tree can quickly be translated to a set of principle of production, and no prior hypothesizes will influence the results generated. However, the downside of using the decision tree algorithm includes incorrectness or imprecision of the decision-making mechanism, and increase on complexity of the model due to more training samples.

3.2. Random Forest

Random forest is an ensemble learning method that builds multiple decision trees independently and combines their predictions through voting (classification) or averaging (regression) to make the final prediction. Prediction using random forests has many advantages, such as not overfitting, proper choice of random type can achieve accurate classification or regression, correlation and strength of predictors can give a good estimate of predictive power, faster than boosting and bagging, a better estimate of internal error, less complicated, and can perform well in parallel processing [6]. Secondly, random forests can take advantage of available features to handle missing data during training and prediction without imputation. However, the random forest also has some drawbacks. Initially, although random forests can perform more accurately than single decision trees, they sacrifice the intrinsic interpretability present in decision trees [8].

3.3. XGBoost

XGBoost (extreme Gradient Boosting) model is a tree-based gradient lifting integrated model with high efficiency and prediction accuracy [4]. XGBoost includes regularization techniques like L1 and L2 regularization, which help prevent overfitting. By controlling model complexity, it generalizes well to unseen data, reducing the risk of overfitting even with high-dimensional feature spaces. Moreover, XGBoost has high speed to optimize for both single- and multi-core processing, which can help to do fast prediction [9]. Although there are some advantages for XGBoost, it also has some disadvantages and limitations that should be taken into consideration. For example, while XGBoost can handle categorical variables, it requires some pre-processing (such as one-hot encoding) to convert them into numerical representations. If not handled properly, this process can result in increased memory usage and possible loss of information.

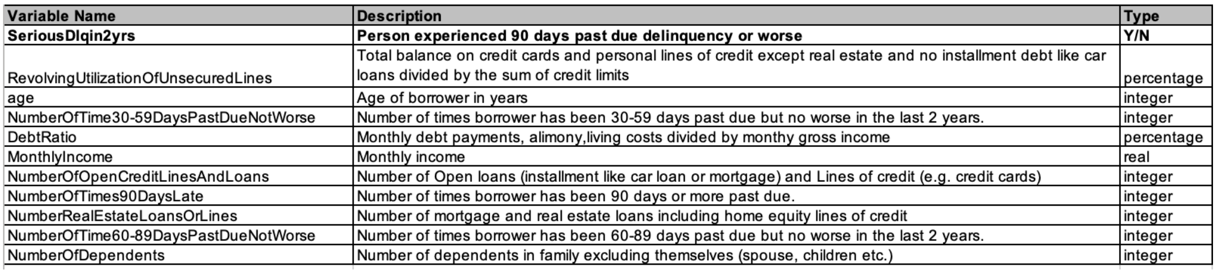
3.4. Convolutional Neural Network

Convolutional neural network, a deep learning algorithm commonly used for image recognition and processing, requires large quantity of unlabeled data for training. It's made up of multiple layers, including the pooling layers, convolutional layers, and fully connected layers. The convolutional layers play a crucial role in determining the neuron outputs by connecting them to localized regions of the input. On the other hand, the pooling layers perform downsampling along the spatial dimensions of the input. Lastly, the fully connected layers generate class scores based on the activations provided, facilitating the classification process. The layers within the CNN consist of neurons organized in three dimensions, encompassing the spatial dimensions of input (height and width) as well as depth. CNN, as a deep learning algorithm, possesses significant predictive capabilities and tends to yield high accuracy when provided with an ample amount of labeled data for training. Nevertheless, the outputs generated by CNN can be challenging for humans to interpret and comprehend [10].

4. Exploratory Data Analysis

4.1. Data Acquisition and Preprocessing

The data we use comes from a kaggle project, It can be downloaded at https://www.kaggle.com/c/GiveMeSomeCredit/overview (Figure 1)



Variable Name	Description	Type
SeriousDlqin2yrs	**Person experienced 90 days past due delinquency or worse**	**Y/N**
RevolvingUtilizationOfUnsecuredLines	Total balance on credit cards and personal lines of credit except real estate and no installment debt like car loans divided by the sum of credit limits	percentage
age	Age of borrower in years	integer
NumberOfTime30-59DaysPastDueNotWorse	Number of times borrower has been 30-59 days past due but no worse in the last 2 years.	integer
DebtRatio	Monthly debt payments, alimony,living costs divided by monthy gross income	percentage
MonthlyIncome	Monthly income	real
NumberOfOpenCreditLinesAndLoans	Number of Open loans (installment like car loan or mortgage) and Lines of credit (e.g. credit cards)	integer
NumberOfTimes90DaysLate	Number of times borrower has been 90 days or more past due.	integer
NumberRealEstateLoansOrLines	Number of mortgage and real estate loans including home equity lines of credit	integer
NumberOfTime60-89DaysPastDueNotWorse	Number of times borrower has been 60-89 days past due but no worse in the last 2 years.	integer
NumberOfDependents	Number of dependents in family excluding themselves (spouse, children etc.)	integer

Figure 1. description of each feature

There are 11 indicators above, among which SeriousDlqin2yrs is the indicator of whether the borrower is overdue, and the remaining 10 indicators are the dependent variables used by this project to judge whether the borrower is overdue. This dataset contains 150,000 rows of data.

Preliminary check for duplicates, we found that a total of 609 rows of data were duplicated

After deleting duplicate values, there are 149391 rows of data remaining

4.2. Data Cleaning

4.2.1. Missing value

By looking at the number of missing values, we found that there are missing values in the features: MonthlyIncome and NumberOfDependents.

For NumberOfDependents, we use KNN to fill missing values.

For MonthlyIncome, we remove missing values.

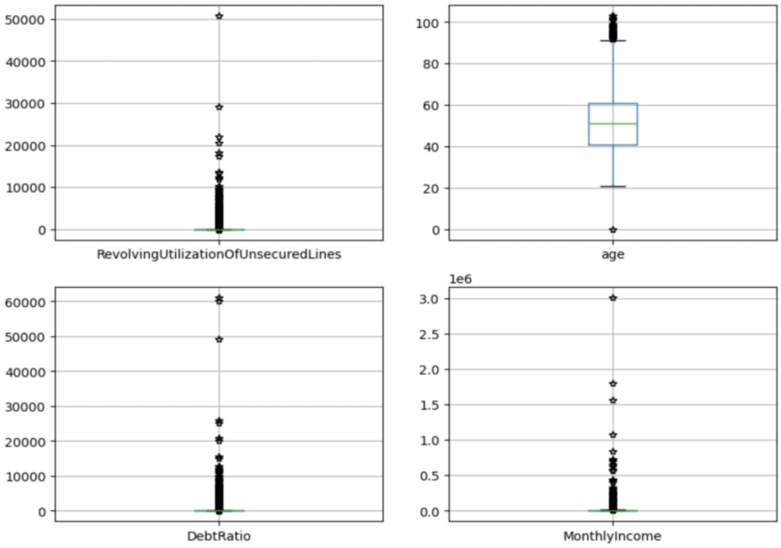
4.2.2. Outliers

After dealing with missing values, we use boxplots to find outliers.

It can be found that the difference between the maximum value and the average value of the RevolvingUtilizationOfUnsecuredLines, DebtRatio, and MonthlyIncome is too large, which is significantly greater than the average value + 3 standard deviations.

Some values of age, NumberOfOpenCreditLinesAndLoans, and NumberOfDependents are greater than the upper limit of the boxplot, but the numerical difference is not too large.

30-59 days, 60-89 days, and 90 days overdue times have obvious outliers (Figure 2).



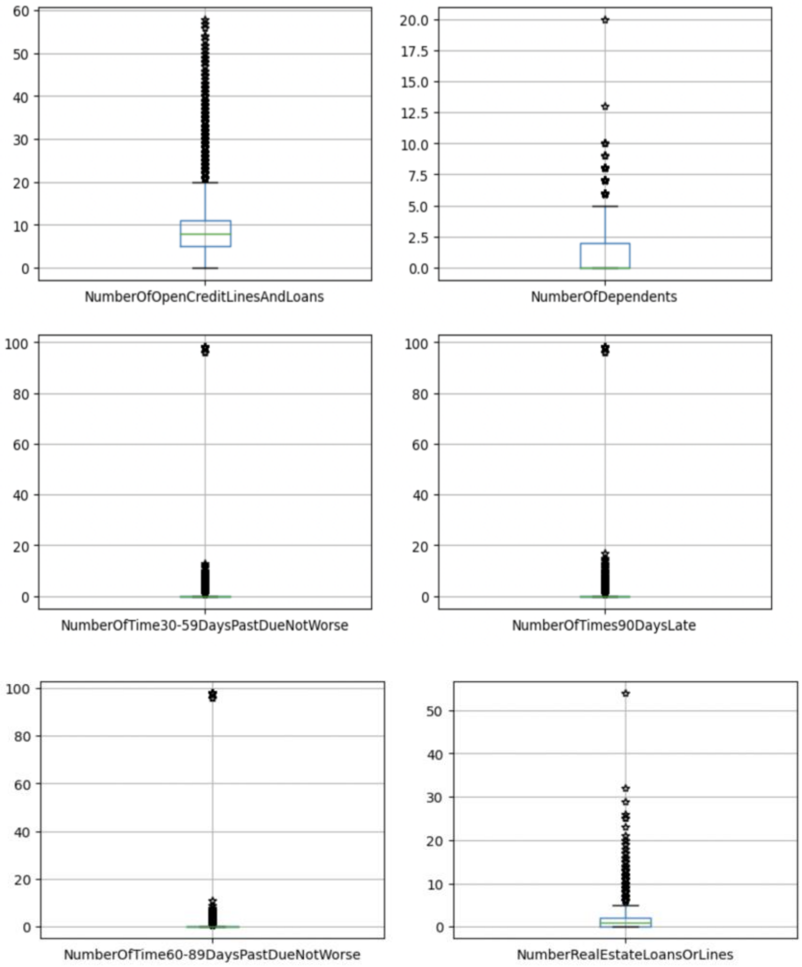


Figure 2. Outliers

4.3. *Outlier Handling Criteria*

Age: delete upper limit value 96, lower limit value 0

RevolvingUtilizationOfUnsecuredLines and DebtRatio: In order to retain the data as much as possible, use the extreme outlier (Q3+3*IQR) calculation of the boxplot to filter. When using Q3+1.5*IQR, the deleted data is too large.

MonthlyIncome: delete values above 500,000

NumberRealestateLoansOrLines: delete values above 20

NumberOfOpenCreditLinesAndLoans: delete values above 20

30-59 days, 60-89 days, 90 days overdue times: delete 98

4.4. *Variable Selection*

In order to prevent linear correlation between variables, we first use Pearson correlation coefficient to delete redundant variables with high correlation (Figure 3). (If there is a correlation coefficient above 0.6, keep one of the variables)

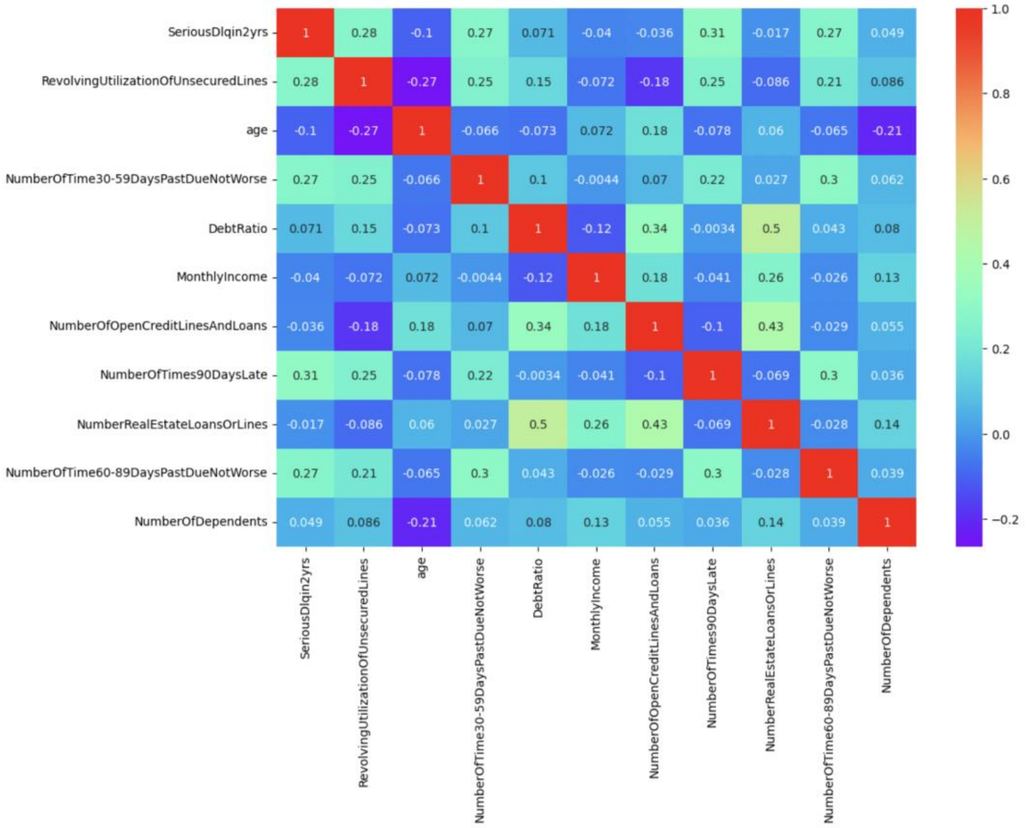


Figure 3. Variable Selection

The correlation coefficients between variables are all below 0.6, so no variables need to be deleted.

4.5. Train-test split

Before dividing the data, look at the proportion of bad samples in the data set, that is 6.9%, which belongs to category imbalance.

The methods to solve the category imbalance problem mainly include undersampling, oversampling, and rescaling.

We choose to take the method undersampling. First divide the training set into two categories: bad samples and good samples, then randomly divide the samples with a large proportion (that is, good samples) into 4 equal parts, and then combine these 4 equal parts of good samples with the bad samples of the training sets respectively, Form 4 new training sets. each of the four training sets is used to train a model.

5. Evaluation Metric

Our model evaluation is mainly based on prediction accuracy, assisted by other metrics, including precision, recall, F1-score, ROC-AUC.

For each model, we output a confusion matrix, a table that summarizes the performance of a classifier which has 4 entries (Table 1):

Table 1. Performance

	Actual Positive	**Actual Negative**
Predicted Positive	True Positives	False Positives

Table 1. (continued).

Predicted Negative	False Negatives	True Negatives

TP refers to the number of samples correctly classified as positive, FP represents the count of samples incorrectly classified as positive, FN denotes the quantity of samples incorrectly classified as negative, and TN signifies the number of samples correctly classified as negative.

Accuracy is the proportion of correctly classified samples among all samples, and is calculated as (TP + TN) / (TP + FP + FN + TN).

Precision is the proportion of true positives among all positive predictions, and is calculated as TP / (TP + FP).

Recall is the proportion of true positives among all actual positive samples, and is calculated as TP / (TP + FN).

F1-score is an alternative machine learning evaluation metric that assesses the predictive skill of a model by elaborating on its class-wise performance. It is the harmonic mean of precision and recall, calculated as 2 * (precision * recall) / (precision + recall). The F1 score takes into account both precision and recall, and provides a balanced measure of the classifier's performance in terms of both false positives and false negatives. It ranges from 0 to 1, where a higher score indicates a better classifier performance in terms of both precision and recall. It is usually used in conjunction with other evaluation metrics, such as accuracy and AUC-ROC.

AUC-ROC (Area Under the Receiver Operating Characteristic curve) measures the trade-off between the true positive rate (TPR) and the false positive rate (FPR) at different decision thresholds, where TPR is the proportion of true positive predictions among all actual positive samples, and FPR is the proportion of false positive predictions among all actual negative samples.

Utilizing the evaluation metrics above, we are able to assess each model's advantages and disadvantages.

6. Experimental Setting

The model setup part was completely conducted on google colab using python.

In our experiment, we attempt to explore the performance of machine learning and deep learning methods in credit risk assessment using a large quantity of data. The dataset we chose contains a limited number of features but a large number of samples.

Due to the high interpretation and simple structure of Decision Tree model, we chose it as the first model. The architecture came from a Python library *DecisionTreeClassifier*.

6.1. XGBoost

We also explored the performance of XGBoost in credit risk assessment using our large quantity dataset. XGBoost, a gradient boosting framework, has garnered significant interest in recent years due to its exceptional accuracy and efficiency. It operates by iteratively incorporating decision trees into the model, effectively minimizing a loss function. The XGBoost algorithm has demonstrated remarkable effectiveness in tackling a diverse array of machine learning tasks including classification problems like credit risk assessment. To implement XGBoost, we used the Python library XGBoost which provides a simple interface for building and training XGBoost models.

6.2. Random Forest

In our experiment, we also assessed the performance of Random Forest in credit risk assessment using our dataset. Random Forest is an ensemble learning technique that combines multiple decision trees to enhance the model's accuracy. It operates by constructing numerous decision trees on random subsets of the data and then aggregating the predictions from these trees. Random Forest has demonstrated its effectiveness in handling high-dimensional datasets with intricate feature interactions, making it a suitable choice for our dataset. To implement Random Forest, we used the Python library RandomForestClassifier, which provides a simple interface for building and training Random Forest models.

6.3. CNN (Convolutional Neural Networks)

Finally, we also explored the performance of Convolutional Neural Networks (CNN) in credit risk assessment using our dataset. In our case, we used a 2D CNN architecture to process the limited number of features in our dataset. The CNN model is made up by multiple convolutional layers and then by max pooling and dense layers for classification. Due to the lack of coding skills, we only managed to manually adjust the parameters. We finally reached a best fitting set of parameters. To implement the CNN model, we used the Keras library in Python, which provides a simple interface for building and training deep learning models.

CNNs have been shown to be very accurate for credit risk assessment. However, it can be more complex to train than other machine learning algorithms, and they require more data samples and features to achieve good performance.

6.4. Grid Search

To achieve optimal performance, we performed grid search hyperparameter tuning for both models. Grid search is widely employed for hyperparameter tuning, where a range of values is specified for each hyperparameter, and the model is trained and evaluated for each combination of hyperparameters in the grid. To perform grid search for Decision Tree, XGBoost, and Random Forest models, we utilized the scikit-learn library in Python. By systematically exploring various hyperparameter combinations, we were able to identify the optimal set of hyperparameters that yielded the best performance on our validation set.

6.5. Hyperparameters

For Decision Trees, the hyperparameters encompass the maximum depth of the tree, the minimum samples required to split a node, the minimum samples required to be at a leaf node, and the criterion used to assess the quality of a split. (e.g., Gini impurity or entropy).

XGBoost: XGBoost's hyperparameters that can be tuned include maximum depth of the trees, regularization parameters, number of trees in the ensemble, the learning rate, and subsampling fraction.

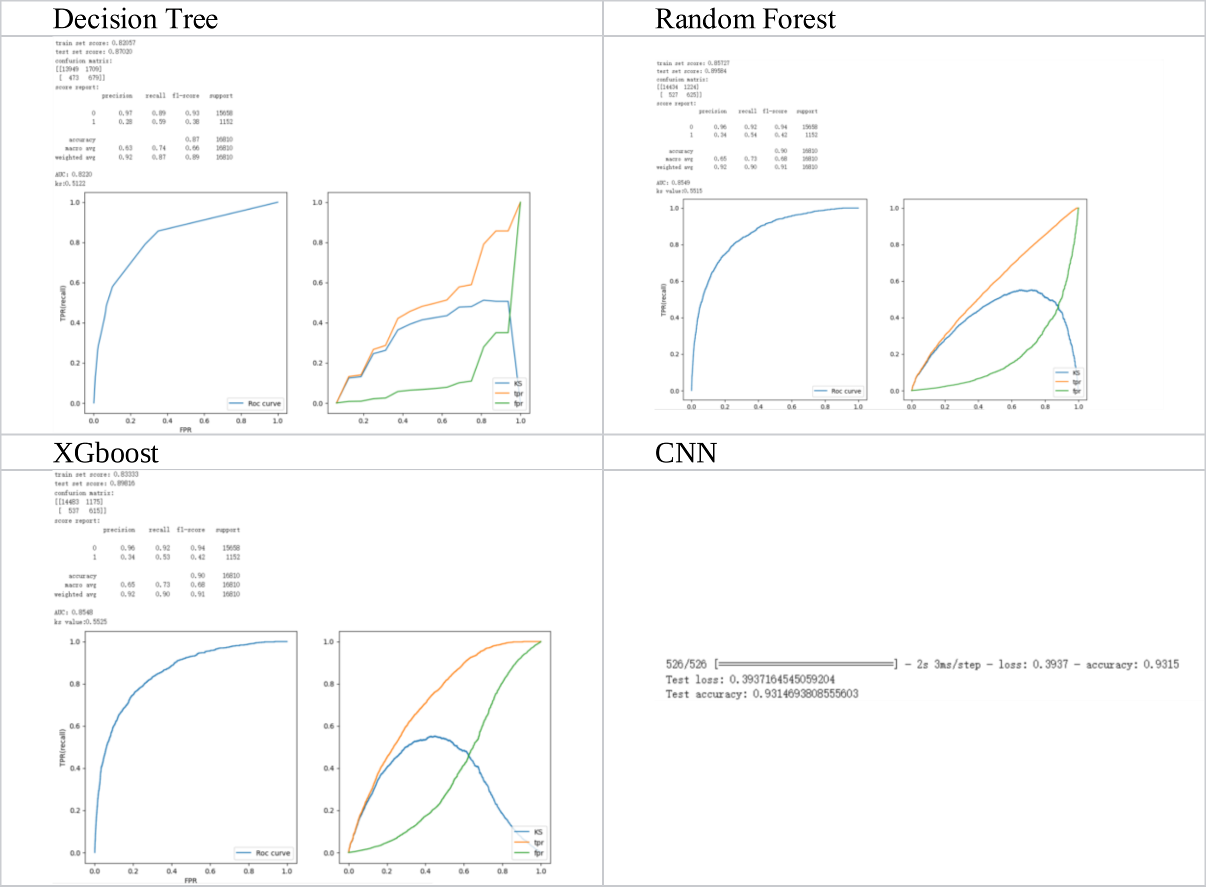
Random Forest: The hyperparameters for Random Forest include the maximum depth of the trees, the number of trees in the ensemble, and the number of features to consider at each split.

CNN: The hyperparameters for a CNN include the number of convolutional layers, the number of filters in each layer, the kernel size of the filters, the activation function used, the number of neurons in the dense layers, the learning rate, and the batch size used during training.

7. Results and Discussion (Limitation)

In our research, we trained four different models on the same dataset to predict credit risk. These models included Decision Trees, Random Forests, XGBoost, and Convolutional Neural Networks (CNN). Through training on the dataset with these four models, we found that each model demonstrated its unique advantages and disadvantages. The Decision Tree model possesses high interpretability and is capable of handling non-linear relationships, but it is prone to overfitting. The more advanced Random Forest, an ensemble learning model, reduces overfitting and enhances accuracy by combining multiple decision trees. However, the Random Forest model requires more time and computational resources during the training process. XGBoost exhibits significant advantages in certain areas. It adopts gradient boosting to optimize the tree model and prevents overfitting through regularization. However, XGBoost's interpretability is not as strong as Decision Trees or Random Forests, and its tuning process can be complex. The Deep Learning model (CNN) performs better in some complex tasks, but in our task, its performance did not exceed the other three machine learning models. The main reason is that although CNN has high accuracy in credit risk, it is a "black box model". Decisions are determined by complex interactions and weights of multilayer neurons, and its complex mechanism is not easily understood by humans, making it extremely poor in interpretability. This is a fatal flaw in the financial industry. Additionally, training deep learning models requires substantial computational resources and time and needs a large amount of data to prevent overfitting. This is not a good choice for the current financial market.

In conclusion, our experimental results confirmed the feasibility and effectiveness of these models in credit risk prediction, but no single model has advantages in all aspects. In practical applications, the model should be developed using the corresponding available resources after analyzing the specific situation. Currently, in the financial market, due to poor interpretability, few individuals or institutions would use CNN to predict credit risk. However, for small online loan company trying to expand their business, CNN might be more suitable. The most likely strategy is to use a learning methodology that allows the strengths of multiple learning modes to be pooled to achieve reasonable accuracy and synchronization.



Decision Tree

```
train set score: 0.82057
test set score: 0.87020
confusion matrix:
[[13949  1709]
 [  473   679]]
score report:
              precision    recall  f1-score   support

           0       0.97      0.89      0.93     15658
           1       0.28      0.59      0.38      1152

    accuracy                           0.87     16810
   macro avg       0.63      0.74      0.66     16810
weighted avg       0.92      0.87      0.89     16810

AUC: 0.8220
ks:0.5122
```

Random Forest

```
train set score: 0.85727
test set score: 0.89584
confusion matrix:
[[14434  1224]
 [  527   625]]
score report:
              precision    recall  f1-score   support

           0       0.96      0.92      0.94     15658
           1       0.34      0.54      0.42      1152

    accuracy                           0.90     16810
   macro avg       0.65      0.73      0.68     16810
weighted avg       0.92      0.90      0.91     16810

AUC: 0.8549
ks value:0.5515
```

XGboost

```
train set score: 0.83333
test set score: 0.89816
confusion matrix:
[[14483  1175]
 [  537   615]]
score report:
              precision    recall  f1-score   support

           0       0.96      0.92      0.94     15658
           1       0.34      0.53      0.42      1152

    accuracy                           0.90     16810
   macro avg       0.65      0.73      0.68     16810
weighted avg       0.92      0.90      0.91     16810

AUC: 0.8548
ks value:0.5525
```

CNN

```
526/526 [==============================] - 2s 3ms/step - loss: 0.3937 - accuracy: 0.9315
Test loss: 0.3937164545059204
Test accuracy: 0.9314693808555603
```

Figure 4. diagram of our results

8. Conclusion

8.1. *Summary*

After training on the dataset using the four models mentioned in this topic, we found that no single model has an absolute overwhelming advantage over the others in predicting credit risk. each model has its own strengths and weaknesses. In practical application scenarios, we should choose the model that best fits the domain to achieve the greatest benefit. For credit risk prediction in this article, using ensemble learning methods to combine the strengths of multiple models for prediction can maximize the benefits.

8.2. Future Work

Due to our current level of coding expertise, we did not fully construct the CNN model in this round of research. The only clear information we have is related to accuracy. Therefore, in our next research experiments, we firstly need to continually improve our coding skills to ensure the smooth and successful training process next time. Then, we need to select datasets with more features. With more features, the model training results will be more accurate and ensure different models under the same dataset won't produce similar results. Subsequently, we will try training more models on our dataset. Only when we have trained a sufficient number of models, can we conveniently and quickly select those that meet our requirements and integrate these models, combining their advantages to maximize the benefits of predicting credit risk. Our ultimate goal is to obtain an ensemble learning model with high prediction accuracy and strong interpretability. This will make a significant contribution to the prediction field in the financial market.

References

[1] Credit risk (2023) Wikipedia. Available at: https://en.wikipedia.org/wiki/Credit_risk (Accessed: 23 July 2023).

[2] Yuejiao, D. et al. (2021) Bank systemic risk around COVID-19: A cross-country analysis, 133. https://doi.org/10.1016/j.jbankfin.2021.106299

[3] Bagchi, B., Chatterjee, S., Ghosh, R., &amp; Dandapat, D. (2020). Impact of covid-19 on Global economy. SpringerBriefs in economics, 15–26. https://doi.org/10.1007/978-981-15-7782-6_3

[4] Wang, K., Li, M., Cheng, J., Zhou, X., &amp; Li, G. (2022). Research on personal credit risk evaluation based on XGBoost. Procedia Computer Science, 199, 1128–1135. https://doi.org/https://doi.org/10.1016/j.procs.2022.01.143

[5] Satchidananda, S. S., & Simha, J. B. (2006). Comparing decision trees with logistic regression for credit risk analysis. International Institute of Information Technology, Bangalore, India. https://www.researchgate.net/profile/S-Satchidananda/publication/237356603_Comparing_decision_trees_with_logistic_regression_for_credit_risk_analysis/links/54dc0ad40cf2a7769d94beff/Comparing-decision-trees-with-logistic-regression-for-credit-risk-analysis.pdf

[6] Ghatasheh, N. (2014). Business analytics using random forest trees for credit risk prediction: A comparison study. International Journal of Advanced Science and Technology, 72, 19–30. https://doi.org/10.14257/ijast.2014.72.02

[7] Charbuty, B., & Abdulazeez, A. (2021, March). Classification based on Decision Tree Algorithm for Machine Learning. Journal of Applied Science and Technology Trends. https://www.jastt.org/index.php/jasttpath/article/view/65

[8] Piryonesi, S. M., & el-Diraby, T. (2021, September 21). Climate change impact on infrastructure: A machine learning solution for predicting pavement condition index. Construction and Building Materials. https://www.sciencedirect.com/science/article/pii/S0950061821026568

[9] Mandal, D. (2023). What are the advantages and disadvantages of XGBoost. https://www.krayonnz.com/user/doubts/detail/623b2b7235e21e005f953106/what-are-the-advantages-and-disadvantages-of-XGBoost

[10] O'Shea, K., & Nash, R. (2015, December 2). An introduction to Convolutional Neural Networks. arXiv.org. https://arxiv.org/abs/1511.08458