Comparative analysis of machine learning models for bond default forecasting based on financial data of Chinese listed companies

Peng Liu, Yuanhang Li*

Jinan University – University of Birmingham Joint Institute, Jinan University, Guangzhou, China ,511443

*Corresponding author: liyuanhang2019051771@stu2019.jnu.edu.cn

Abstract. At a time when bond defaults become frequent and market confidence is undermined, the subject of how to accurately predict bond defaults merits investigation. This paper combined popular machine learning methods to predict bond defaults by selecting financial data of 23 listed companies in China that defaulted on credit bonds from 2013 to 2022 and 230 financial data of listed companies that did not default on credit bonds during the same period, using logistic regression, random forest, support vector machine (SVM), and K Nearest Neighbors (KNN) to estimate the probability of bond defaults by listed companies and compare the predictive performance of these methods. The financial data are found to be quite good at predicting bond defaults of listed companies, and the SVM model performs the best.

Keywords: Bond Default Forecasting, Machine Learning, Chinese Listed Companies Financial Data.

1. Introduction

In recent years, against the background of further complicated and severe international situation, frequent occurrences of epidemics, and increased economic downward pressure, the number of bond defaults in China has reached record highs, with a defaulted bond size of 140.9 billion yuan[1] in 2021. The high default risk rate largely affects investors' confidence in bond investment, which in turn will affect enterprises' financing through bond channels. Therefore, it is important to predict bond defaults more accurately. Nowadays, machine learning, as the main research direction in the field of artificial intelligence, has demonstrated its advantages in processing a largamountsnt of data efficiently and predicting accurately in many fields, and how to take advantage of machine learning in the field of finance is a widely discussed topic. This paper combines machine learning algorithms to analyze the financial data of bond defaulting companies and tries to use the financial data to predict whether bond defaults will occur in listed companies through machine learning algorithms. In this paper, after aggregating the financial data of Chinese listed companies with bond default records between 2013 and 2022, we use Logistic Regression, Random Forest, Support Vector Machine (SVM), KNearest Neighbors (KNN) to predict the default of credit bonds issued by Chinese listed companies. We also compare the prediction results of the above algorithms and their generalization ability.

1.1 Data collection

For collecting default data, this paper aggregates a total of 259 defaults in all credit bonds of non-financial companies from December 31, 2013 to July 31, 2022. Considering that the financial data of listed companies can be more easily obtained from annual reports and are accurate and reliable, the default data of non-listed companies are excluded in this paper. And after removing the listed companies with missing data due to delisting, 23 default data were left, which corresponded to the financial cross-section data in the annual reports of 23 listed companies in the year of first default.

Similarly, for non-default data selection, the financial cross-section data of 230 non-default companies in the annual reports of the year of first default are selected in this paper. In this paper, we randomly select the listed companies with complete financial data and without debt default under the SWS L2 Industry Classification to which the defaulted companies belong, and determine the number of similar companies selected according to 1:10. [1] If there are not enough 10 comparable companies

in the SWS L2 Industry Classification, the data of companies under the SWS L1 Industry Classification of are used.

According to the time of default of their bonds, the following financial data of the annual reports of the corresponding listed companies in that year were selected in table 1.[2]

Table 1. Financial indicators

	Financial indicators	Calculation
Liquidity & Solvency Ratios	Current Ratio	Current assets/current liabilities
	Quick Ratio	(Current assets - net inventories)/current liabilities
	Non-Current Asset to Total Asset Ratio	Non-current assets/total assets
	Debt to Asset Ratio	Total liabilities/total assets
Efficiency Ratios	Account Receivable Turnover	Operating income/[(beginning of period net notes and accounts receivable + ending net notes and accounts receivable)/2]
	Inventory Turnover	Operating costs/[(Beginning Inventory + Ending Inventory)/2]
	Total Asset Turnover	Total operating revenue/[(beginning total assets+ ending total assets)/2]
	All Amount Invested in Capital	Total shareholders' equity + Total liabilities - Non-interest-bearing current liabilities - Non-interest-bearing long-term liabilities
	Working Capital	Current assets - current liabilities
Profitability Ratios	ROE	Net profit attributable to shareholders of the parent/[(Equity attributable to the parent at the beginning of the period + equity attributable to shareholders of the parent at the end of the period)/2]*100%
	ROE(TTM)	Net profit attributable to shareholders of the parent/[(Equity attributable to the parent at the beginning of the period + equity attributable to shareholders of the parent at the end of the period)/2]*100%
	ROA	EBIT / Total Assets * 100%
	ROA(TTM)	EBIT / Total Assets * 100%
	EPS	Net profit for the period attributable to ordinary shareholders/weighted average number of ordinary shares actually issued and outstanding during the period
	Return Earning	Surplus + undistributed profit

1.2 Default data descriptive statistics

Since a default of the same listed company is likely to occur in multiple bonds during the same period, while this is primarily due to the listed firm's bad business condition. If we discuss the number of defaulted bonds, it is possible that we may double-count different bonds issued by the same company, resulting in data that are biased towards the year the company defaulted and the industry that it operates in. Consequently, we are doing statistical analysis on the number of default companies as opposed to the number of default bonds. By analyzing the time of the first default and the industry to which the defaulted company belongs, we discovered the following:

Firstly, Table 2 reveals that the number of defaulted listed companies from 2018 to 2022 is on average higher than the number of defaulted listed companies from 2014 to 2017, with 2018 having the highest number of defaulted listed companies.

Second, from Table3, it is evident that the real estate development industry classification has much more defaulting companies than the other industry classes. The next major industries are Diversified Finance, Chinese Medicine, and Rare Metals.

Table 2. Time of Default

Time of Default	2014	2015	2016	2017	2018	2019	2020	2021	2022	Total
Number of companies	1	3	0	1	6	5	4	4	1	25

Table 3. Industry to which the default company belongs

Industry to which the default company belongs	Number of companies
Real Estate Development	5
Diversified Finance	2
Chinese Medicine	2
Rare Metals	2
Commercial Property Management	1
Computer Applications	1
Specialized equipment	1
General Machinery	1
Gas	1
Food & Beverage	1
Electricity	1
Power supply equipment	1
Landscaping	1
Optical Optoelectronics	1
Specialty Retail	1
Chemical products	1
Packaging Printing	1
Environmental Engineering & Services	1
Total	25

1.3 Data processing

In the original data, the number of defaulted bonds is 23, while the number of non-defaulted bonds is 230. the small volume of defaulted category data and the large gap between the number of defaulted and non-defaulted companies cause a serious category data imbalance problem, which is likely to cause the wrong evaluation of the model. To address this problem, this paper employs the Synthetic Minority Oversampling Technique (SMOTE) sampling method [3], and the ENN data cleaning

algorithm [4] for the original dataset. The original dataset was processed by the SMOTE-ENN algorithm and the data type ratio was adjusted to be close to 1:1.[5]

1.3.1 SMOTE Algorithm

The SMOTE algorithm is a combination of oversampling and undersampling sampling method, i.e., randomly increasing the number of samples from the minority class and randomly decreasing the number of samples from a majority class. It is based on the k nearest neighbor sample points of each sample point, and randomly selects N neighbor points for difference multiplication by a threshold in the range of [0,1] to synthesize the data. The core of this algorithm is that the neighboring points on the feature space have similar features. It does not sample on the data space, but in the feature space, so its accuracy will be higher than the traditional sampling method. [3][6][7] However, if the selected minority class samples are surrounded by majority class samples, and such samples may be noisy, the newly synthesized samples will have most overlap with the surrounding majority class samples, leading to classification difficulties. Therefore, to avoid this problem, this paper further uses the ENN algorithm to clean the data.

1.3.2 ENN Algorithm

The principle of ENN algorithm is to find the k nearest neighboring points of a certain sample, and if more than half of the k points are similar to the sample then keep the re-sample, and vice versa delete the re-sample. the ENN algorithm solves the shortcomings of SMOTE algorithm and better solves the sample imbalance problem. [5]

2. Analysis methods and evaluation matrix

In this paper, the following four machine learning methods will be used to identify potential default bonds.

2.1 Logistic Regression

Logistic regression is a machine learning method used to solve a binary classification problem to estimate the likelihood of something happening. Logistic regression is widely used in industry because of its strong explanatory power. The task in this paper is to identify defaulted bonds, which is a typical binary classification problem, and thus can be solved using logistic regression. [8]

$$h_\theta(x) = g_\theta(\theta^T x) \tag{1}$$

$$g(z) = \frac{1}{1+e^{-z}} \tag{2}$$

$$h_\theta(x) = \frac{1}{1+e^{-\theta^T x}} \tag{3}$$

In (1)(2)(3), x is the financial indicator characteristics of the company and θ is the parameter corresponding to the financial indicator.

$$p(y = 1 | x; \theta) = h_\theta(x) = \frac{1}{1+e^{-\theta^T x}} \tag{4}$$

$$\hat{y} = 1, if\ h_\theta(x) > threshold \tag{5}$$

When the value of $h_\theta(x)$ is greater than the set threshold, it can be predicted as the bond is a default bond. To ensure that the logistic regression is more sensitive to the identification of defaulted bonds, the threshold is lowered from 0.5 to 0.35 in this paper. [8]

2.2 Random Forest

Random forest algorithm builds multiple decision trees and combines them to get a more accurate and stable model, which is a combination of bagging idea and random selection feature. Random forest constructs multiple decision trees, and when a sample is to be predicted, the prediction results of each tree in the forest are counted, and then the final result is selected from these predictions by voting. Each tree in the random forest is split according to a node when it is generated, and the split can be based on the reduction of Gini coefficients before and after the split. We use the sum of the reduction of Gini coefficients of each tree in the random forest split according to feature m (m is a randomly selected feature) as a criterion to judge the importance of feature m.

$$Gini(p) = \sum_{k=1}^{K} p_k(1 - p_k) \tag{6}$$

There will be 1/3 of the data set not used in the random forest algorithm, out of bag (OOB), so there is no need to divide the test set and training set additionally in the random forest model, and just use OOB as the test set. [9]

2.3 Support Vector Machine (SVM)

In other classifiers, the resulting decision boundary is at a small distance from the nearest data point. When a new data is at the decision boundary, the classifier may cause misclassification due to some noise effects. In contrast, for a support vector machine, if the data points are p-dimensional vectors, the SVM uses a p-1 dimensional hyperplane to separate these points and chooses the hyperplane that separates the two classes at the maximum interval. A hyperplane can be determined by the normal vector W and the intercept b, whose equation is

$$X^T W + b = 0 \tag{7}$$

where X is the feature column vector. To find the maximum interval hyperplane, we can first choose two parallel hyperplanes (i.e., $X^T W + b = 1$ and $X^T W + b = -1$) that separate the two types of data so that the distance between them is as large as possible. The area within these two hyperplanes is called the "margin", and the maximum interval hyperplane is the hyperplane located directly between them.

$$margin = \rho = \frac{2}{||W||} \tag{8}$$

To make margin maximal, i.e., ρ maximal, equivalently, ρ^2 maximal.

$$\max_{W,p} \rho \leftrightarrow \max_{W,p} \rho^2 \leftrightarrow \min_{W,p} \frac{1}{2}||W||^2 \tag{9}$$

Adding the constraint yields.

$$\begin{gathered} \min_{W,p} \frac{1}{2}||W||^2 \\ s.t\ y_i(X_i^T W + b) \geq 1\ , i = 1, \dots, n \end{gathered} \tag{10}$$

By solving the above optimization problem, the optimal hyperplane is obtained. [10]

2.4 K Nearest Neighbors (KNN)

In the KNN algorithm, given a point, the categorization is done by observing the distance (this paper uses Euclidean distance) to the label of the nearest K points of the point. the most critical

parameter in the KNN algorithm is K. [10]This paper tried several values of K and finally found that the algorithm works best when K = 5.

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + \cdots + (x_n - y_n)^2} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{11}$$

2.5 Evaluation matrix (Evaluation)

2.5.1 Accuracy

The proportion of the correctly predicted samples to the total samples, the greater the accuracy rate, the better.

$$Accuracy = \frac{TP+FN}{TP+TN+FP+FN} \tag{12}$$

2.5.2 Recall

The proportion of positive samples predicted from the actual positive samples. In the problem of bond default identification, what matters more is the accuracy of the actual defaulted bonds, so the size of the observed recall rate can be a good judge of the model's performance.

$$Recall = \frac{TP}{TP+FP} \tag{13}$$

2.5.3 Precision

The percentage of the predicted positive samples that are actually positive. By observing the percentage, we can judge whether the algorithm is too sensitive to the defaulted bonds. Especially in the logistic regression algorithm, the threshold size can be adjusted according to the precision rate.

$$Precision = \frac{TP}{TP+TN} \tag{14}$$

2.5.4 F1-Score

The summation average of the perfection rate and recall rate. The perfection rate and recall rate affect each other, and although a high rate of both is an ideal situation, in practice it is often the case that a high precision rate is associated with a low recall rate, or a low recall rate is associated with a high precision rate. If a balance of both is needed, then the F1-Score can be used.

$$F1 = \frac{Precision*Recall}{Precision+Recall} \tag{15}$$

3. Summary of results.

In this paper, four models, logistic regression, Random Forest, KNN(K=5), and SVM, are used to train and test the bond default data. In the sample set, 80% of the defaulted bond data and non-defaulted bond data are randomly selected to form the training set, and the rest is the test set. The 4-Fold cross-validation (Random Forest model using OBB instead of K-Fold) was also performed in the training set, and the results were obtained as shown in Table 4.

Table 4. Summary of results

		Accuracy	Recall	Precision	F1-Score
SVM	Training Set	1	1	1	1
	Cross-Validation Set	1	1	1	1
	Test set	1	1	1	1
KNN, K=5	Training Set	1	1	1	1
	Cross-Validation Set	0.986	0.986	0.986	0.986
	Test set	0.991	0.991	0.991	0.991
Random Forest	Training Set	1	1	1	1
	OOB	0.968	/	/	/
	Test set	0.991	0.991	0.991	0.991
Logistc Regression	Training Set	0.861	0.861	0.89	0.857
	Cross-Validation Set	/	/	/	/
	Test set	0.827	0.827	0.879	0.828

Among the four models, the SVM model is the most effective, achieving perfect classification in both the training and test sets. The KNN algorithm and Random Forest algorithm have the same effect, with the F1score of 0.991 in the test set, while the Logistic model is less effective, with an F1score of 0.828.

In addition to the above results, the Random Forest model can also derive the importance of characteristics. From Figure 1, we can find that EPS, ROE, and ROA are the most important. Therefore, a rough estimate of whether a bond will default or not can be made by looking at the high and low levels of these three indicators

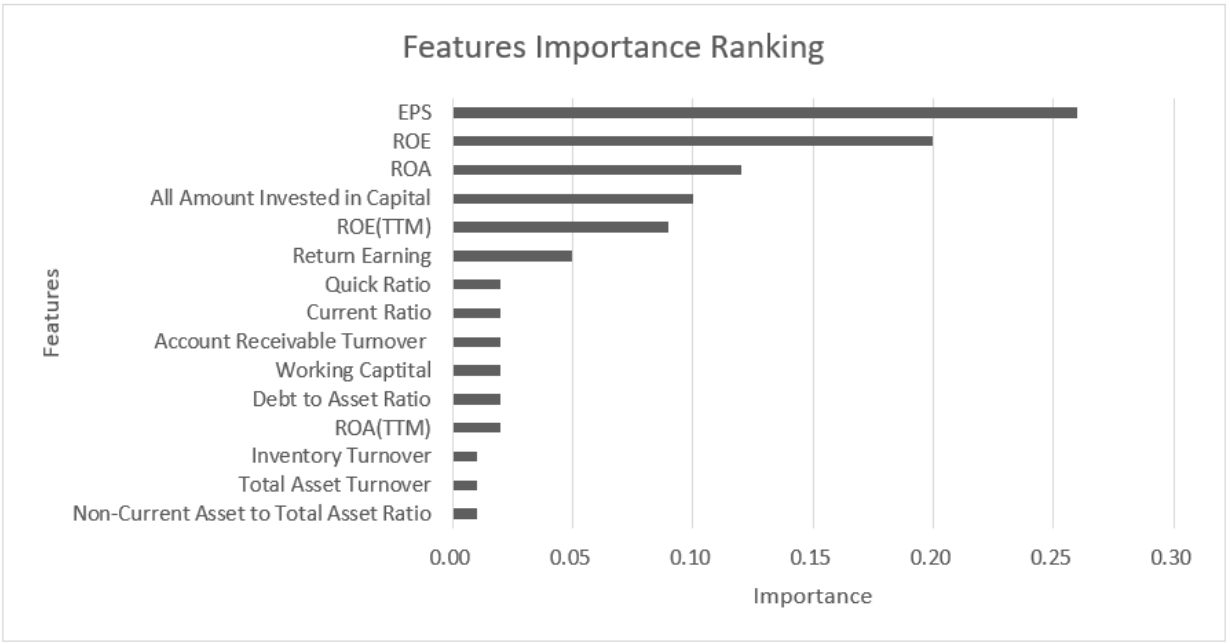

Figure 1. Features Importance Ranking

4. Conclusions

This paper used Logistic Regression, Random Forest, Support Vector Machine and K Nearest Neighbors (KNN) to predict the default of credit bonds issued by Chinese listed companies based on the financial data of Chinese listed companies. It is found that the financial data of Chinese listed companies are excellent in predicting bonds, and the SVM algorithm, KNN algorithm, and Random Forest algorithm are found to perform better than logistic regression in default prediction, which

supports the further research on the selection of algorithms for machine learning in bond default prediction.

References

[1] DAI Yarong & SHEN Yifeng. (2022). Can the Random Forest Model Predict Boond Default in China?. China Journal of Econometrics (02),418-440.

[2] CHEN Ziyang. (2020). A study of financial irregularities in listed companies based on decision trees and random forest algorithms. Contemporary Finance (07),23-25+16.

[3] Fernandez, A., Garcia, S., Herrera, F., & Chawla, N. (2018). SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. *Journal Of Artificial Intelligence Research*, *61*, 863-905. doi: 10.1613/jair.1.11192

[4] B. Tang and H. He, "ENN: Extended Nearest Neighbor Method for Pattern Recognition [Research Frontier]," in IEEE Computational Intelligence Magazine, vol. 10, no. 3, pp. 52-60, Aug. 2015, doi: 10.1109/MCI.2015.2437512.

[5] Guan, H., Zhang, Y., Xian, M., Cheng, H., & Tang, X. (2020). SMOTE-WENN: Solving class imbalance and small sample problems by oversampling and distance scaling. *Applied Intelligence*, *51*(3), 1394-1409. doi: 10.1007/s10489-020-01852-8

[6] H. He and E. A. Garcia, "Learning from Imbalanced Data," in IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 9, pp. 1263-1284, Sept. 2009, doi: 10.1109/TKDE.2008.239.

[7] J. Wang, M. Xu, H. Wang and J. Zhang, "Classification of Imbalanced Data by Using the SMOTE Algorithm and Locally Linear Embedding," 2006 8th international Conference on Signal Processing, 2006, pp. , doi: 10.1109/ICOSP.2006.345752.

[8] Li, P., Zhou, R., & Xiong, Y. (2020). Can ESG Performance Affect Bond Default Rate? Evidence from China. *Sustainability*, *12*(7), 2954. doi: 10.3390/su12072954

[9] Zhu, L., Qiu, D., Ergu, D., Ying, C., & Liu, K. (2019). A study on predicting loan default based on the random forest algorithm. *Procedia Computer Science*, *162*, 503-513. doi: 10.1016/j.procs.2019.12.017

[10] Liang, D., Lu, C., Tsai, C., & Shih, G. (2016). Financial ratios and corporate governance indicators in bankruptcy prediction: A comprehensive study. *European Journal Of Operational Research*, *252*(2), 561-572. doi: 10.1016/j.ejor.2016.01.012