Received: 5 December 2022 | Revised: 15 January 2024 | Accepted: 18 January 2024

DOI: 10.1002/for.3074

RESEARCH ARTICLE

WILEY

Two-stage credit risk prediction framework based on three-way decisions with automatic threshold learning

Yusheng Li[1] | **Mengyi Sha**[2]

[1]School of Economics and Management, Hebei University of Technology, Tianjin, China

[2]Department of Civil Engineering, McGill University, Montreal, Quebec, Canada

Correspondence
Yusheng Li, School of Economics and Management, Hebei University of Technology, no. 8, Guangrong Road, Dingzigu Street, Hongqiao District, Tianjin 300130, China.
Email: yusheng.li@hebut.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Number: 72301093; Natural Science Foundation of Hebei Province, Grant/Award Number: G2023202004; Science and Technology Project of Hebei Education Department, Grant/Award Number: QN2023121

Abstract

Credit risk prediction is a binary classification problem. Using two-way decisions to classify defaulters may lead to decision errors due to insufficient information. To solve this issue, in addition to identifying borrowers as defaulters and nondefaulters, this paper introduced the delay-decision mechanism in three-way decisions, so that records acquiring more information do not make decisions immediately. A two-stage credit risk prediction framework based on three-way decisions was proposed to reduce decision risk. In this framework, the decision cost values of three-way decisions were simplified by analyzing the credit risk prediction, and the expression of threshold calculation was also modified. An optimization objective was built according to the trade-off between information gain and decision cost, and the particle swarm optimization algorithm was applied to learn the decision thresholds. After adding more supplementary information, the samples in the delayed-decision region were made further decisions. A dataset from a commercial bank in China was employed to conduct experiments, and the results demonstrated that our proposed method outperformed various base classifiers.

KEYWORDS

credit risk prediction, information gain, particle swarm optimization, three-way decisions, thresholds learning

1 | INTRODUCTION

Lending growth by financial companies and banks drives a country's economic growth and reflects the state of the economy (Tariq et al., 2019). If there is no proper credit risk management, financial institutions might suffer huge losses because of credit default (Zhang et al., 2018). The loan market is substantially growing in many countries; for example, according to data released by the People's Bank of China, as of the end of June 2022 in China, the balance of RMB loans was 206.4 trillion yuan, a year-on-year increase of 11.2%. With the expansion of the loan business, financial institutions face increasing credit risk pressures in lending, there is evidence that the outstanding business credit of Canada, America, and England was $2,262 billion, $15,243 billion, and £18,582 billion in 2019, respectively (Machado & Karray, 2022). Therefore, it is necessary to establish a credit risk assessment model to effectively reduce possible economic losses.

The credit risk prediction is to judge the risk of borrowers not being able to pay on time by their characteristics. The research can be divided into two aspects: identifying the risk factors affecting credit default or designing algorithms to predict credit risk. For the risk factors of credit default, both hard information and soft information can affect the default status (J. W. Lee et al., 2021). Hard information reflects the economic and social situation of the borrower, such as personal income, credit ratings, marital status, education status, and family members. Soft information is nonfinancial information that is not easy to obtain, such as social media information (Ge et al., 2017), and some scholars integrated text-related soft information to evaluate credit risk (Liang & He, 2020; Xia et al., 2019). For algorithms of credit risk prediction, machine learning methods have been widely used in recent years (Fu et al., 2020), which makes it possible for credit risk analysts to use big data technologies to make credit decisions. Some supervised methods had been applied in credit risk prediction, such as logistic regression (LR) (Dumitrescu et al., 2022), K-nearest neighbors (KNN) (T. Wang et al., 2022), decision trees (DT) (Zhou et al., 2017), neural network (NN) (Oreski et al., 2012; Xia et al., 2021), support vector machine (SVM) (X. Xu et al., 2009), deep neural networks (CNN) (Liu et al., 2022; M. Yang et al., 2023) and so on, and some unsupervised methods were used to evaluate credit risk, such as hierarchical clustering (Zeitsch, 2019), k-means (Kou et al., 2014; Song et al., 2020), and self-organizing maps (SOM) (Bao et al., 2019; L. Wang et al., 2020). Because supervised methods perform better and financial institutions can accumulate some labeled default data in many cases, supervised algorithms are more widely used. These studies aimed to improve the identification of credit defaults, and most of them adopt a two-way decision approach; that is, each record would be uniquely judged as default or nondefault. However, some records cannot immediately make decisions in the current information (Maldonado et al., 2020); supplementary information is needed to assist detection, and immediately making a classification may cause decision errors.

To solve the above problem, credit default prediction needs to apply three-way decisions (TWD), and delayed decision-making is required for records that cannot be decided immediately. The three-way decision is a decision-making model related to human cognition, and the concept was proposed by Yao according to the rough set of decision theory (Yao, 2009). It divides a unified set into three regions: positive region (POS), negative region (NEG), and boundary region (BND). POS and NEG are certainty regions, including records that can make decisions immediately, and the decision strategies they represent are acceptance and rejection. BND is an uncertain region; including records that cannot give an immediate decision under the existing information, its decision strategy is delayed decision. The division of regions relies on thresholds α and β. Yao connected practical management decisions with threshold setting and applied Bayesian decision theory and decision-theoretic rough set to minimize the decision cost for threshold learning (Deng & Yao, 2014; J. Xu et al., 2020). However, the threshold calculation relies on a predetermined cost matrix for different decision costs whose values are difficult to quantify. Therefore, the determination of the three-way decision thresholds is a difficult problem.

Regarding the calculation of three decision thresholds, many researchers designed different optimization objectives to realize the learning of thresholds. Jia et al. (2014) used thresholds to present loss functions, and the minimization of decision cost was taken as the objective function to obtain the thresholds. Multi-objective optimization methods were also proposed. In addition to decision cost, Pan et al. (2016) used the difference in thresholds as another optimization objective to limit the range of the BND region. Li and Shao constructed an optimization objective based on the decision cost and the uncertainty of BND area to determine three-way decision thresholds in crime linkage (Li & Shao, 2022). Shen et al. took the maximization of information gain after regional division as the optimization goal to learn the thresholds (Shen et al., 2022). To reduce the number of loss functions, Suo et al. re-expressed thresholds by a compensation coefficient, and just one parameter needs to be preset in the three-way decision (Suo, Tao, Zhu, Chen, et al., 2020; Suo, Tao, Zhu, Miao, et al., 2020). When learning the three-way decision thresholds, these studies design optimization objectives from the decision cost and the statistical characteristics of samples in different regions. On this basis, this paper proposes a threshold-setting method from the above two points considering the actual situation of credit risk.

The objective of this study is to decrease the classification errors incurred by two-way decisions in credit risk prediction. This paper introduces three-way decisions in credit risk prediction, and some records do not make immediate decisions but make decisions after additional information is provided. For the threshold learning in three-way decisions, we construct an optimization objective according to the information gain and the overall decision cost and formulate the three-way decision rules. Therefore, our contributions mainly include the following twofolds:

1. A novel two-stage credit risk prediction framework incorporating three-way decision theory (TSCR) is established. The first stage uses algorithms to evaluate the default probability of records, and TWD divides the records into POS, NEG, and BND regions. In the second stage, the prediction results of samples in the POS and NEG areas are taken as supplementary information to complete the classification of the BND region.
2. The method for automatic learning of three-way decision thresholds is designed. The goal is to make the information gain after threshold division high and the

overall decision cost low. The objective function is optimized by the particle swarm optimization (PSO) algorithm to learn thresholds.

This paper has five sections; the rest are as follows. Section 2 provides briefly introduces the relevant theories. Section 3 details the framework of credit risk prediction with three-way decisions and the learning of thresholds. Experiments are conducted on a loan default dataset and results are analyzed in Section 4. Section 5 provides the conclusion.

2 | THEORETICAL BACKGROUND

2.1 | Three-way decisions

The three-way decision is a kind of soft computing model (Yue et al., 2020); its generic idea is to divide the universe into three disjoint regions: POS, BND, and NEG, which involve three decision rules: acceptance, delayed decisions, and rejection, respectively.

The division process of regions in three-way decisions is as follows (Fang et al., 2020): Given three decision actions $\{a_P, a_B, a_N\}$ means an entity is classified into POS, BND, or NEG region. Suppose each entity has two states $\{X^C, X^{\neg C}\}$ and $Pr(X^C|x)$ is the conditional probability when the entity x is in the state X^C, so $P_r(X^C|x) + P_r(X^{\neg C}|x) = 1$. In different states, taking an action will have different decision costs, as shown in Table 1. If the state is X^C, λ_{PP}, λ_{BP}, and λ_{NP} mean the losses of selecting a_P, a_B, and a_N actions. The losses of selecting a_P, a_B, and a_N actions are λ_{PN}, λ_{BN}, and λ_{NN} when the state is $X^{\neg C}$. According to Table 1, the expected loss of an entity after taking different decision actions can be expressed as follows:

$$\begin{aligned} R(a_P|x) &= \lambda_{PP} \cdot P_r(X^C|x) + \lambda_{PN} \cdot P_r(X^{\neg C}|x), \\ R(a_B|x) &= \lambda_{BP} \cdot P_r(X^C|x) + \lambda_{BN} \cdot P_r(X^{\neg C}|x), \\ R(a_N|x) &= \lambda_{NP} \cdot P_r(X^C|x) + \lambda_{NN} \cdot P_r(X^{\neg C}|x) \end{aligned} \tag{1}$$

where $R(a_P|x)$, $R(a_B|x)$, and $R(a_N|x)$ are related to the conditional probability and decision cost. TWD will take

TABLE 1 The loss function matrix in three-way decisions.

Action		State	
		X^C	$X^{\neg C}$
Acceptance	a_P	λ_{PP}	λ_{PN}
Rejection	a_B	λ_{BP}	λ_{BN}
Delay decision	a_N	λ_{NP}	λ_{NN}

the action with the least decision cost; therefore, the decision rules can be summarized as follows:

(P) If $R(a_P|x) \le R(a_B|x)$ and $R(a_P|x) \le R(a_N|x)$, then $x \in POS(X)$;
(B) If $R(a_B|x) \le R(a_P|x)$ and $R(a_B|x) \le R(a_N|x)$, then $x \in BND(X)$;
(N) If $R(a_N|x) \le R(a_P|x)$ and $R(a_N|x) \le R(a_B|x)$, then $x \in NEG(X)$;
If loss functions meet the following condition(J. Yang et al., 2019; Yao, 2010):

$$\lambda_{PP} \le \lambda_{BP} < \lambda_{NP}, \lambda_{NN} \le \lambda_{BN} < \lambda_{PN} \tag{2}$$

$$\frac{\lambda_{NP} - \lambda_{BP}}{\lambda_{BN} - \lambda_{NN}} > \frac{\lambda_{BN} - \lambda_{NN}}{\lambda_{PN} - \lambda_{BN}} \tag{3}$$

The decision rules can be simplified as follows:

(P1) If $P_r(X^C|x) \ge \alpha$, then $x \in POS(X)$;
(N1) If $P_r(X^C|x) \le \beta$, then $x \in NEG(X)$;
(B1) If $\beta < P_r(X^C|x) < \alpha$, then $x \in BND(X)$.

where α and β are calculated by Equation (4).

$$\alpha = \frac{(\lambda_{PN} - \lambda_{BN})}{(\lambda_{PN} - \lambda_{BN}) + (\lambda_{BP} - \lambda_{PP})}, \beta = \frac{(\lambda_{BN} - \lambda_{NN})}{(\lambda_{BN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{BP})} \tag{4}$$

2.2 | PSO algorithm

The PSO algorithm is a parallel evolutionary computation technique proposed by Eberhart and Kennedy in 1995 (Eberhart & Kennedy, 1995). Its advantages are fast convergence and easy implementation (C.-S. Lee et al., 2018). In PSO, the position of each particle is a candidate solution to the problem. The quality of a particle depends on the fitness value of the particle's position in space. The position $x_i(t)$ of i th particle in this generation is determined by its position $x_i(t-1)$ in the previous generation and its velocity vector v_i, and its velocity decides the direction and distance of each flight of the particle (Dziwinski & Bartczuk, 2020).

At the beginning of PSO, some particles are initialized first, and their positions and velocities are random. Each particle is randomly assigned a speed and iteratively changes its position at that speed. In each iteration, the particle updates itself by tracking the best historical position $xbest_i$ of itself and the best global position $gbest$ of all particles; the formula is as follows:

$$x_i(t) = x_i(t-1) + v_i(t) \tag{5}$$

$$v_i(t) = \omega \cdot v_i(t-1) + \psi_1 \cdot r_1 \cdot (xbest_i - x_i(t-1)) + \psi_2 \cdot r_2 \cdot (gbest - x_i(t-1)) \tag{6}$$

The position and velocity both need to be updated. ω is the inertia weight and its range is $(0,1]$; it controls the effect of the previous step's velocity $v_i(t-1)$ on $v_i(t)$. ψ_1 and ψ_2 are acceleration coefficients, which are constant and refer to the cognitive parameter and social parameter, respectively. r_1 and r_2 are random numbers on the uniform distribution in the range of $(0,1]$. $xbest_i$ and $gbest$ are determined according to the fitness of the particle position.

3 | THREE-WAY DECISIONS CLASSIFICATION IN CREDIT RISK

In this section, we will introduce the two-stage credit risk prediction framework (TSCR). The role of three-way decisions in TSCR and the learning of thresholds will also be described in detail.

3.1 | Classification framework of credit risk with TWD

This paper applies the three-way decisions to credit risk prediction and proposes a two-stage classification framework. The basic process is shown in Figure 1.

The first stage completes the initial classification of the dataset. The labeled data are the input of classification algorithms as the training set, and the first-stage classification model Model-1 is obtained. Model-1 estimates the default probability of records, and the threshold pair (α, β) is obtained to split the unlabeled data.

The second stage classifies samples in the BND area. The POS and NEG areas divided by the three-way decisions are regarded as certain samples, and labels are assigned to them, which are fused with labeled data as additional information to form supplementary labeled data. Model-2 will be trained on the supplementary

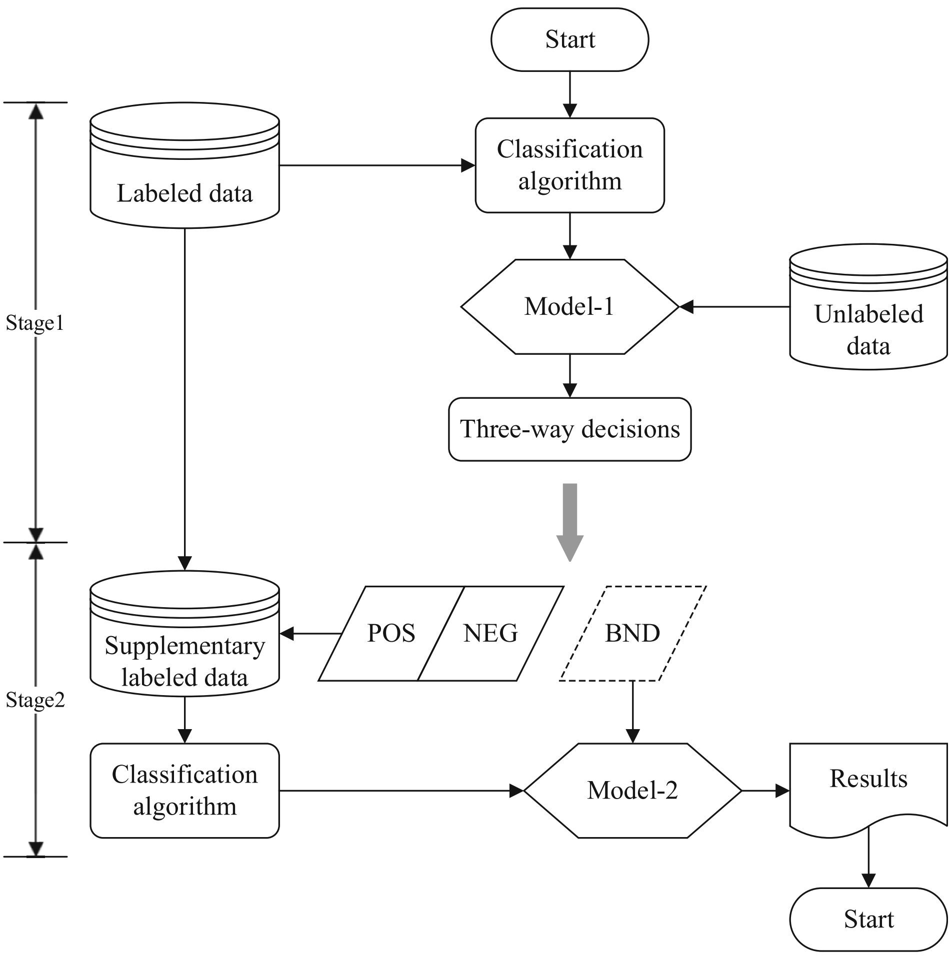

FIGURE 1 The framework of two-stage credit risk prediction with three-way decisions.

labeled data. The classification result of samples in the BND region is given by Model-2.

3.2 | Thresholds learning for the TWD

In Section 3.1, three-way decisions divide the data into different regions. When thresholds are used to split regions, samples in different regions are subsets of the universe. The thresholds are learned by information gain and decision cost. In this section, we introduce the optimization objective of threshold learning.

3.2.1 | Information gain in TWD

In information theory, the greater the information entropy, the higher the uncertainty of a random variable or classification region. The information gain means the degree to which the information complexity (uncertainty) is reduced under a condition. In credit risk prediction, information gain can represent the degree to which the uncertainty of the default record can be reduced under the division of the threshold pair (α,β). The entropy of the dataset U is calculated as follows:

$$H(U) = -\frac{|X^C|}{|U|}\log\frac{|X^C|}{|U|} - \frac{|X^{\neg C}|}{|U|}\log\frac{|X^{\neg C}|}{|U|} \tag{7}$$

Under the division of the threshold pair (α,β), $NEG(X)$ is the set of samples in the NEG area, and the information entropy of the NEG region is defined as follows:

$$H_{NEG}(\alpha,\beta) = -\frac{|NEG(X)\cap X^C|}{|NEG(X)|}\log\frac{|NEG(X)\cap X^C|}{|NEG(X)|} - \frac{|NEG(X)\cap X^{\neg C}|}{|NEG(X)|}\log\frac{|NEG(X)\cap X^{\neg C}|}{|NEG(X)|} \tag{8}$$

where $\frac{|NEG(X)\cap X^C|}{|NEG(X)|}$ represents the ratio of samples in NEG region belonging to the state C, $\frac{|NEG(X)\cap X^{\neg C}|}{|NEG(X)|}$ represents the ratio of samples in NEG region belonging to the state $\neg C$, and their sum is 1. Relative to the universal set U, the conditional probabilities of NEG, POS, and BND are as follows:

$$P_{NEG} = \frac{|NEG(X)|}{|U|},\ P_{POS} = \frac{|POS(X)|}{|U|},\ P_{BND} = \frac{|BND(X)|}{|U|} \tag{9}$$

According to the overall entropy of the universe and the entropy of different regions, the information gain of credit risk prediction after applying the three-way decisions is calculated as follows:

$$IGain(\alpha,\beta) = H(U) - P_{POS}H_{POS}(\alpha,\beta) - P_{NEG}H_{NEG}(\alpha,\beta) - P_{BND}H_{BND}(\alpha,\beta) \tag{10}$$

when the number of samples in state X^C and $X^{\neg C}$ is the same, that is, $\frac{|X^C|}{|U|} = \frac{|X^{\neg C}|}{|U|} = 0.5$, $H(U)$ gets the maximum value of 1.0. After the universe is divided into three regions, the higher the purity of the sample category in each region, the smaller the entropy value after division, and the information gain will be greater.

3.2.2 | Decision cost of TWD

Table 1 involves six loss functions; we can first simplify the loss matrix by analyzing the credit risk decision. λ_{PP} and λ_{NN} are the loss of correctly classifying default and nondefault records. Obviously, the loss of correct classification can be ignored, so we set $\lambda_{PP} = \lambda_{NN} = 0$. In addition, whether a default record or nondefault record is divided into the BND area, a further decision is required after obtaining supplementary information, so the decision costs of λ_{BP} and λ_{BN} are the same, so we set $\lambda_{BP} = \lambda_{BN} = \lambda_B$. For misclassified records, the decision cost is higher than that of correct classification and the BND region records, that is, $\lambda_{PN} > \lambda_B > 0$, $\lambda_{NP} > \lambda_B > 0$. The simplified cost matrix is shown in Table 2, and there are only λ_{PN}, λ_{NP}, and λ_B three unknown loss functions.

Based on Table 2, the expected decision cost for three-way decisions Equation (1) can be re-expressed as follows:

$$\begin{aligned} R(a_P|x) &= \lambda_{PN}\cdot P_r\left(X^{\neg C}|x\right), \\ R(a_B|x) &= \lambda_B\cdot P_r\left(X^C|x\right) + \lambda_B\cdot P_r\left(X^{\neg C}|x\right) = \lambda_B, \\ R(a_N|x) &= \lambda_{NP}\cdot P_r\left(X^C|x\right) \end{aligned} \tag{11}$$

TABLE 2 The simplified loss function matrix in credit risk prediction.

	Default	Nondefault
a_P	0	λ_{PN}
a_B	λ_B	λ_B
a_N	λ_{NP}	0

where $P_r(X^C|x) + P_r(X^{\neg C}|x) = 1$, so $R(a_B|x) = \lambda_B$. The calculation of α and β in Equation (4) can also be simplified as follows:

$$\alpha = \frac{(\lambda_{PN} - \lambda_B)}{(\lambda_{PN} - \lambda_B) + (\lambda_B - 0)} = \frac{\lambda_{PN} - \lambda_B}{\lambda_{PN}}, \quad \beta = \frac{(\lambda_{NP} - 0)}{(\lambda_{NP} - 0) + (\lambda_{NP} - \lambda_B)} = \frac{\lambda_B}{\lambda_{NP}} \tag{12}$$

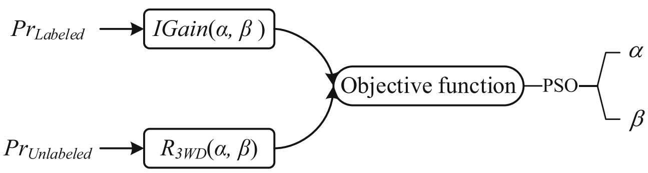

FIGURE 2 The objective function of TWD with automatic threshold learning.

According to Equation (12), the calculation of λ_{PN} and λ_{NP} can be re-expressed by α, β, and λ_B (see Equation (13)).

$$\lambda_{PN} = \frac{\lambda_B}{1-\alpha}, \lambda_{NP} = \frac{\lambda_B}{\beta} \tag{13}$$

In Equation (13), λ_{PN}, λ_{NP}, and λ_B establish relations. If λ_B is set to 1, λ_{PN} and λ_{NP} can be directly associated with α and β:

$$\lambda_{PN} = \frac{1}{1-\alpha}, \lambda_{NP} = \frac{1}{\beta} \tag{14}$$

The decision cost of each sample is obtained according to Equation (11). The decision cost of a region is the sum of the cost of samples falling into the region. Because regions are divided by α and β, so the decision

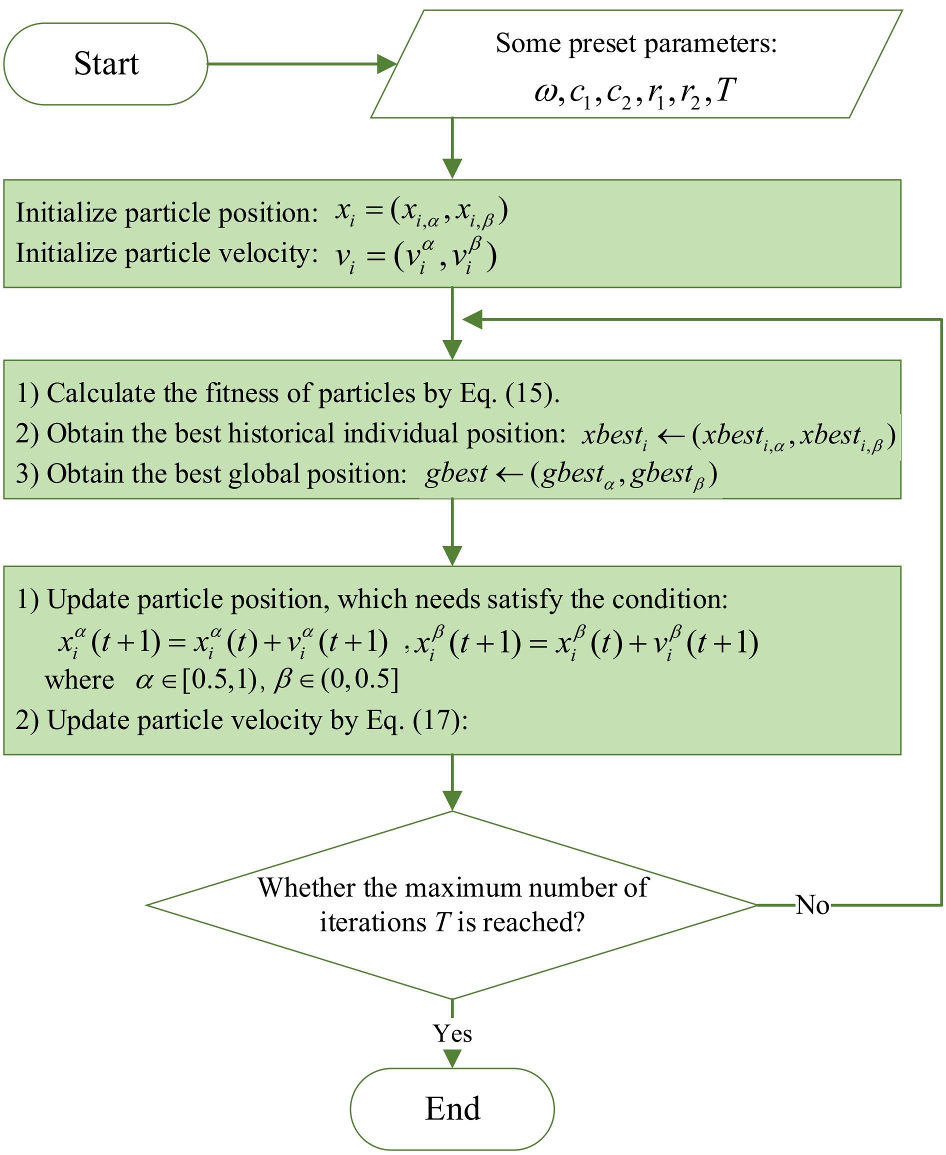

FIGURE 3 Flowchart of learning thresholds by particle swarm optimization (PSO) algorithm.

TABLE 3 Features and their descriptions.

Feature	Description	Type
total_loan	Amount of loan	Continuous
year_of_loan	Year of loan	Discrete
interest	Current lending rate	Continuous
monthly_payment	Installment amount	Continuous
class	Loan grade	Categorical
work_type	Type of job	Categorical
employment_type	Company type	Categorical
industry	Field of work	Categorical
work_year	Working years	Continuous
house_ownership	Whether the borrower owns a house	Categorical
house_loan_status	Status of home loan	Categorical
marriage	Status of marriage	Categorical
offsprings	Status of offsprings	Categorical
issue_date	Month of loan disbursement	Categorical
use	Loan purpose category	Categorical
post_code	Borrower's zip code	Continuous
region	Code of district	Discrete
debt_loan_ratio	Debt-to-income ratio	Discrete
del_in_18month	Number of default events by borrowers more than 30 days past due in the past 18 months	Continuous
early_return	Number of prepayments	Continuous
early_return_amount	Accumulated amount of prepayment	Continuous
early_return_amount_3mon	Amount of early repayment in the past 3 months	Continuous
title	The loan name provided by the borrower	Categorical
f0-f5	Anonymous feature	Continuous

cost of POS, BND, and NEG regions can be calculated as follows.

$$\begin{aligned}
R_{POS} &= \sum_{P_r(X^C|x)>\alpha} R(a_P|x) = \sum_{x\in POS(X)} R(a_P|x),\\
R_{BND} &= \sum_{\beta<P_r(X^C|x)\le\alpha} R(a_B|x) = \sum_{x\in BND(X)} R(a_B|x),\\
R_{NEG} &= \sum_{P_r(X^C|x)\le\beta} R(a_N|x) = \sum_{x\in NEG(X)} R(a_N|x)
\end{aligned} \tag{15}$$

The overall decision cost is composed of the decision costs of POS, BND, and NEG regions, as shown in Equation (16).

$$R_{3WD}(\alpha,\beta) = R_{POS} + R_{BND} + R_{NEG} \tag{16}$$

3.2.3 | Objective function

The samples in the training set are labeled, and the algorithm can calculate the conditional probability of labeled samples. Given a pair of thresholds (α,β), it divided the data set into different regions according to three-way decision rules. Greater information gain after region division means this threshold pair is better. The algorithm can also calculate the conditional probability of unlabeled samples. After applying three-way decisions, the smaller the decision cost of unlabeled samples, the better the pair of thresholds. Based on information gain and overall decision cost, the trade-off can be taken as the optimization objective (Figure 2); it is defined as Equation (17):

$$\underset{\alpha,\beta}{argMax} : Z = IGain(\alpha,\beta)/R_{3WD}(\alpha,\beta) \tag{17}$$

Because α and β are not discrete, it is impossible to learn the optimal thresholds in an exhaustive way, and we exploit heuristic algorithms PSO to obtain the approximate solution. The basic process of PSO to find the optimal thresholds is as follows. In the beginning, the position of each particle is denoted by a threshold pair $x_i = \left(x_i^\alpha, x_i^\beta\right)$. x_i^α and v_i^α are the position and velocity of i th particle in the α dimension. The particle position and velocity in tth step are updated by Equations (18) and (19), respectively.

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{18}$$

$$v_i^d(t+1) = \omega \cdot v_i^d(t) + \psi_1 \cdot r_1 \cdot \left(xbest_i^d - x_i^d(t)\right) + \psi_2 \cdot r_2 \cdot \left(gbest^d - x_i^d(t)\right) \quad (19)$$

where d is α or β. $x_i^d(t)$ means the d dimension position of ith particle in the tth step. $xbest_i^d$ is the best historical position of the particle x_i in d dimension. $gbest^d$ is the best global position of all particles in d dimension. The optimization objective Equation (17) is the calculation formula for the fitness of particles.

The learning process of thresholds is shown in Figure 3. First, particles and some parameters are initialized. Second, the information gain, decision cost, and optimization goal (the fitness of particles) in different particle positions are calculated. The best historical individual position $xbest_{i,d}$ and the best global position $gbest_d$ are recorded. Thirdly, the position and velocity of particles will be updated. It should be noted that the position of a particle is constrained because $\alpha \in [0.5, 1)$ and $\beta \in (0, 0.5]$. When the number of iterations reaches the maximum T, the algorithm terminates.

4 | EMPIRICAL STUDY

In the experiment, we evaluate the effectiveness of the TSCR framework on real datasets. First, we show the information of the data set and introduce the indicators for evaluating the algorithm's performance. Then, we describe the settings of the experiment, including algorithms and parameter settings. Finally, the experimental results are investigated and analyzed.

4.1 | Data set

The experiments are carried out on a personal loan dataset of a local corporate bank in Henan province, China. The features and their descriptions of the dataset are shown in Table 3, and there are continuous, discrete, and categorical features. In this section, we construct three datasets of different scales with 20,000, 50,000, and 100,000 records, respectively. The number of default data and nondefault data in each dataset is shown in Table 4.

4.2 | Evaluation indicators

A confusion matrix can be used as the basis of many evaluation indicators. The confusion matrix is in Table 5. True positive (TP) is the number of correctly classifying default records. True negative (TN) is the number of correctly classifying nondefault records. False negative (FN) is the number of incorrectly classifying default records. False positive (FP) is the number of incorrectly classifying nondefault records.

Based on the confusion matrix, the following index can be calculated. True positive rate (TPR) refers to the ratio of correctly classifying default records, and true negative rate (TNR) refers to the ratio of correctly classifying nondefault records.

$$TPR = TP/(TP + FN) \quad (20)$$

$$TNR = TN/(TN + FP) \quad (21)$$

In addition, the following metrics are used for evaluation. Precision measures the correct rate of finding default records, and Recall represents the ratio of all default records that are correctly found. F-measure (FM) and G-measure (GM) are two comprehensive evaluation metrics, and they more comprehensively reflect the performance of algorithms.

$$Precision = TP/(TP + FP) \quad (22)$$

$$Recall = TP/(TP + FN) \quad (23)$$

$$FM = \frac{(1 + b^2) \times Recall \times Precision}{b^2 \times Recall \times Precision} \quad (24)$$

TABLE 5 The confusion matrix.

Confusion matrix		Predict: Default	Nondefault	Total
Actual	Default	*TP*	*FN*	*P*
	Nondefault	*FP*	*TN*	*N*
	Total	*P′*	*F′*	*P + N*

TABLE 4 Datasets and their information.

Dataset	Number of default records	Number of nondefault records	Total number
D20000	3937	16,063	20,000
D50000	9831	40,169	50,000
D100000	19,465	80,535	100,000

TABLE 6 Parameters of PSO.

Parameter	Population	Iteration	ω	ψ_1	ψ_2
Value	200	50	[0.4, 0.9]	2	2

Abbreviation: PSO, particle swarm optimization.

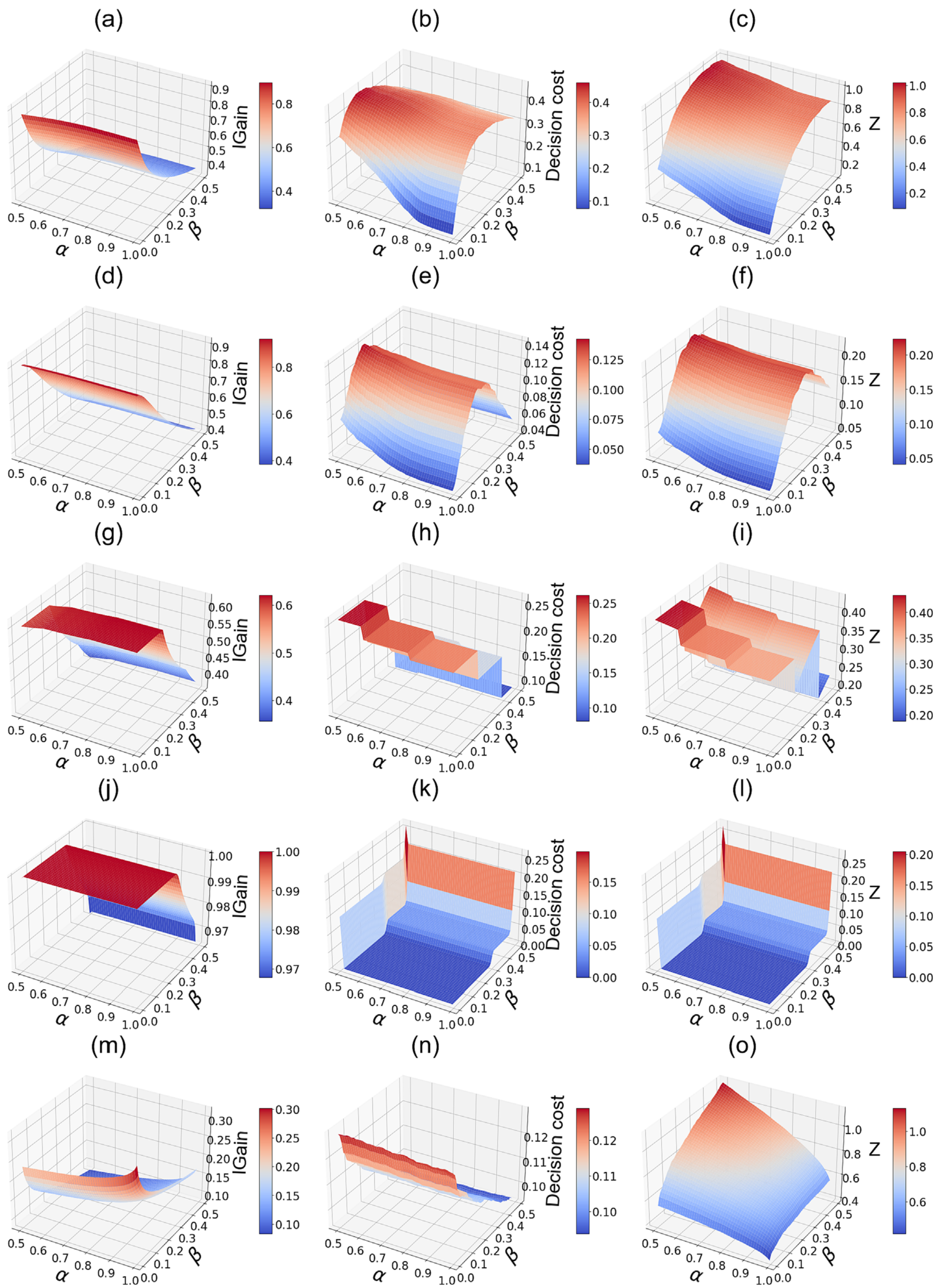

FIGURE 4 The IGain, decision cost, and Z with the change of and β.

TABLE 7 Results comparison of base algorithms and the first stage of TSCR.

		TPR		TNR		FM		GM		
Algorithm	**Dataset**	**2W**	**TSCR1**	**2W**	**TSCR1**	**2W**	**TSCR1**	**2W**	**TSCR1**	**rB**
Adaboost	D20000	.4201	.8409	.9414	.9116	.5058	**.7246**	.6286	**.8754**	.368
KNN	D20000	.1062	.2720	.9371	.8645	.1556	**.2812**	.3146	**.4838**	.550
LGBM	D20000	.4395	.5030	.9376	.9351	.5183	**.5601**	.6416	**.6854**	.057
LR	D20000	.2042	.2566	.9693	.9727	.3053	**.3378**	.4428	**.4465**	.227
NN	D20000	.3312	.3312	.8471	.8471	**.2709**	.2708	**.4109**	.4107	.002
Adaboost	D50000	.4349	.8298	.9405	.9144	.5181	**.7234**	.6394	**.8710**	.338
KNN	D50000	.1122	.2832	.9352	.8628	.1629	**.2901**	.3237	**.4940**	.543
LGBM	D50000	.4497	.5299	.9386	.9356	.5287	**.5802**	.6496	**.7039**	.068
LR	D50000	.2318	.2847	.9660	.9698	.3377	**.3603**	**.4726**	.4641	.227
NN	D50000	.4079	.4080	.8100	.8101	.3119	**.3120**	**.4657**	**.4657**	.001
Adaboost	D100000	.4408	.8325	.9427	.9172	.5253	**.7300**	.6446	**.8737**	.339
KNN	D100000	.1180	.2940	.9356	.8681	.1704	**.3002**	.3321	**.5051**	.529
LGBM	D100000	.4601	.5447	.9407	.9376	.5395	**.5935**	.6579	**.7145**	.070
LR	D100000	.2347	.2443	.9662	.9741	**.3410**	.3068	**.4757**	.3934	.238
NN	D100000	.3347	.3346	.8683	.8684	**.2875**	.2872	**.4344**	.4339	.002

Abbreviations: KNN, K-nearest neighbors; LGBM, light gradient boosting machine; LR, logistic regression; NN, neural network; TNR, true negative rate; TPR, true positive rate; TSCR, three-way decision theory.

TABLE 8 Results comparison of base algorithms and TSCR2.

		TPR		TNR		FM		GM	
Algorithm	**Dataset**	**2W**	**TSCR2**	**2W**	**TSCR2**	**2W**	**TSCR2**	**2W**	**TSCR2**
Adaboost	D20000	.4201	.4481	.9414	.9347	.5058	**.5222**	.6286	**.6469**
KNN	D20000	.1062	.1177	.9371	.9292	.1556	**.1670**	.3146	**.3298**
LGBM	D20000	.4395	.4772	.9376	.9286	.5183	**.5391**	.6416	**.6654**
LR	D20000	.2042	.2260	.9693	.9643	.3053	**.3283**	.4428	**.4655**
NN	D20000	.3312	.3327	.8471	.8466	.2709	**.2734**	.4109	**.4170**
Adaboost	D50000	.4349	.4590	.9405	.9349	.5181	**.5318**	.6394	**.6549**
KNN	D50000	.1122	.1241	.9352	.9272	.1629	**.1745**	.3237	**.3390**
LGBM	D50000	.4497	.4839	.9386	.9301	.5287	**.5468**	.6496	**.6708**
LR	D50000	.2318	.2484	.9660	.9620	.3377	**.3533**	.4726	**.4882**
NN	D50000	.4079	.4091	.8100	.8099	.3119	**.3138**	.4657	**.4686**
Adaboost	D100000	.4408	.4661	.9427	.9368	.5253	**.5395**	.6446	**.6607**
KNN	D100000	.1180	.1310	.9356	.9278	.1704	**.1832**	.3321	**.3485**
LGBM	D100000	.4601	.4884	.9407	.9334	.5395	**.5537**	.6579	**.6751**
LR	D100000	.2347	.2475	.9662	.9632	.3410	**.3529**	.4757	**.4876**
NN	D100000	.3347	.3369	.8683	.8680	.2875	**.2906**	.4344	**.4385**

Abbreviations: KNN, K-nearest neighbors; LGBM, light gradient boosting machine; LR, logistic regression; NN, neural network; TNR, true negative rate; TPR, true positive rate; TSCR, three-way decision theory.

$$GM = \sqrt{\frac{TP}{TP+FN} \times \frac{TN}{TN+FP}} \tag{25}$$

4.3 | Experimental setting

The experiment applies 10-fold cross-validation; the dataset is divided into 10 equal parts, 9 of which are used as the training set, and the remaining 1 is the test set. We repeat 10 times cross-validation, and the cross-validation results are averaged. This paper uses Adaboost, KNN, light gradient boosting machine (LGBM), LR, and NN to predict loan defaults. The implementation of the algorithm is completed in Python language. The PSO algorithm is used to solve the optimization objective. For the parameter setting of PSO, we refer to the literature (Shi & Eberhart, 1999), and parameters are set in Table 6.

4.4 | Results and analysis

In the TSCR framework, the objective function Z is designed to obtain the thresholds of TWD. To observe the changes in information gain, decision cost, and Z value, we record their values under different threshold pairs (α,

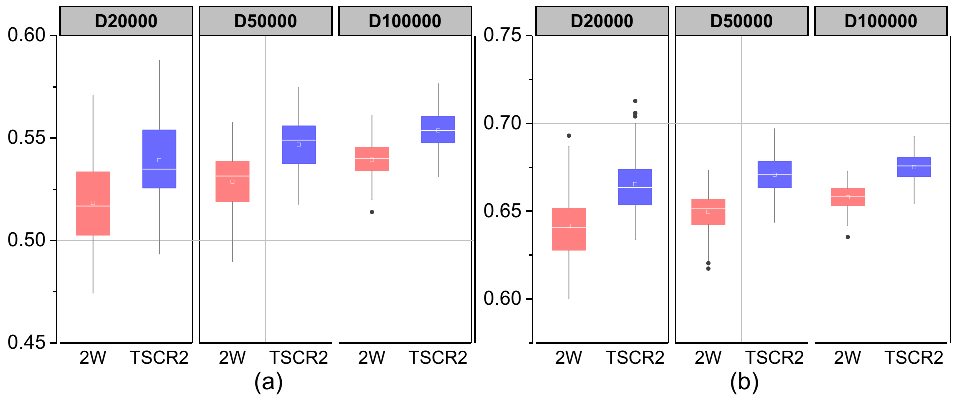

FIGURE 5 The boxplot of 2W and TSCR2 in FM and GM. TSCR, three-way decision theory.

TABLE 9 Results comparison of base algorithm and TSCR3.

		TPR		TNR		FM		GM	
Algorithm	Dataset	2W	TSCR3	2W	TSCR3	2W	TSCR3	2W	TSCR3
Adaboost	D20000	.4201	.9204	.9414	.9414	.5058	**.8522**	.6286	**.9308**
KNN	D20000	.1062	.7160	.9371	.9371	.1556	**.7256**	.3146	**.8191**
LGBM	D20000	.4395	.5654	.9376	.9378	.5183	**.6208**	.6416	**.7277**
LR	D20000	.2042	.5968	.9693	.9765	.3053	**.6984**	.4428	**.7591**
NN	D20000	.3312	.3344	.8471	.8473	.2709	**.2760**	.4109	**.4205**
Adaboost	D50000	.4349	.9108	.9405	.9405	.5181	**.8456**	.6394	**.9255**
KNN	D50000	.1122	.7160	.9352	.9352	.1629	**.7229**	.3237	**.8183**
LGBM	D50000	.4497	.6002	.9386	.9387	.5287	**.6481**	.6496	**.7503**
LR	D50000	.2318	.6122	.9660	.9739	.3377	**.7072**	.4726	**.7687**
NN	D50000	.4079	.4095	.8100	.8101	.3119	**.3209**	.4657	**.4693**
Adaboost	D100000	.4408	.9111	.9427	.9427	.5253	**.8482**	.6446	**.9268**
KNN	D100000	.1180	.7167	.9356	.9356	.1704	**.7227**	.3321	**.8189**
LGBM	D100000	.4601	.6150	.9407	.9407	.5395	**.6610**	.6579	**.7605**
LR	D100000	.2347	.6110	.9662	.9776	.3410	**.7085**	.4757	**.7672**
NN	D100000	.3347	.3400	.8683	.8685	.2875	**.2959**	.4344	**.4458**

Abbreviations: KNN, K-nearest neighbors; LGBM, light gradient boosting machine; LR, logistic regression; NN, neural network; TNR, true negative rate; TPR, true positive rate; TSCR, three-way decision theory.

β), and the results are shown in Figure 4. The dataset D20000 was used for this experiment, and the decision cost was averaged to better display the results. The value ranges of α and β were $[0.5,1)$ and $(0,0.5]$, respectively. In Figure 4, the decision cost and information gain reached the optimal value under different threshold pairs (α, β), and the Z value reached the maximum in different regions.

In the first stage of TSCR, three-way decisions classify records into POS, NEG, and BND areas, and samples in the BND region do not make decisions. 2W means the base algorithm adopts two-way decisions. The results comparison between 2W and the first stage of TSCR (TSCR1) is shown in Table 7, and better results are bold. The rB refers to the ratio of samples in the BND area in the total samples; for example, 0.368

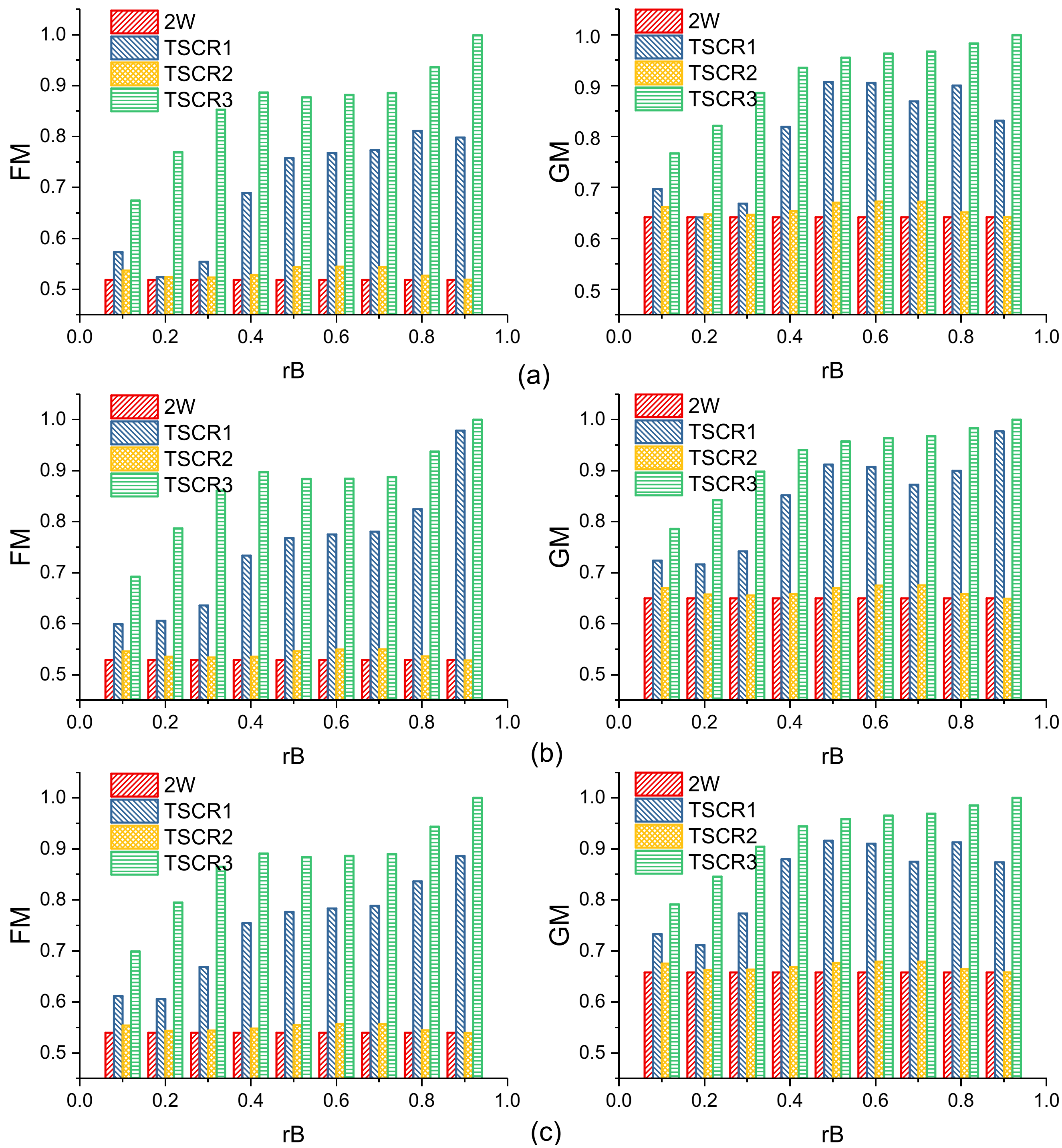

FIGURE 6 The FM and GM of LGBM in different rB.

means 36.8% of samples are divided into the BND region. Because α is greater than 0.5 and β is less than 0.5, samples in the BND area are difficult to classify. In Table 7, the FM and GM are improved a lot in most cases since samples of the BND region are not classified in the first stage, and the improvement may be more obvious if the rB is higher.

In the second stage of the TSCR, BND regional samples are classified, and the results are shown in Table 8, and better results are bold. After completing the second stage classification TSCR (TSCR2), compared with the base algorithm, the classification results are improved in FM and GM indicators, and the maximum improvement is about 2.4%. Among these algorithms, LGBM has the best classification result, so we take it as an example to make the boxplot of 2W and TSCR in terms of FM and GM (see Figure 5). In Figure 5, the boxplot of TSCR is higher than that of 2W, and TSCR is more centralized, which indicates that TSCR is more stable and can achieve better performance.

In the second stage of TSCR, the classified samples are added as supplementary information to train a new classification model, and samples of the BND area are classified in the second stage. If the supplementary information is the knowledge of credit analysts, that is, the samples of the BND region are not classified by algorithms but are judged manually by analysts, then they may have different performances. The samples of the BND area are only a part of the universal set, if the number of records requiring manual processing is acceptable, then it can be considered for processing by credit analysts, so we also analyze the classification results in this case. We assume that the credit analysts are all experts and can correctly find the defaulter. The results comparison of 2W and TSCR with expert (TSCR3) are shown in Table 9, and better results are bold. Because the samples of the BND area that are difficult to classify are judged by experts, who provide more professional decision-making information, better performance can be achieved after the first-stage classification. Manual processing is more time-consuming and costly, so only if the number of samples in the BND area is small, they can be handled manually.

In the TSCR framework, there are no restrictions on the ratio of samples in the BND region. We also investigate the classification performance in different rB when the base algorithm was LGBM. In Figure 6, with the increase of rB, the performance of TSCR2 was close to LGBM in two-way decisions. The FM and GM of TSCR1 and TSCR3 are gradually improved with the increase of rB, because more and more samples are not classified by algorithms, and they are left or are judged by experts. In practical management, the number of samples in the BND region can also be controlled. If financial institutions intend to let the analyst handle samples in the BND region, they can choose a moderate proportion rB.

5 | CONCLUSION

In credit risk prediction, sometimes it is difficult to immediately give a judgment on whether the borrower would default in the future. If a decision is forcibly made, it may cause decision errors. Therefore, we introduced the three-way decisions; in addition to identifying borrowers as defaulters or nondefaulters, it added a BND region where samples did not require immediate decision-making. Based on three-way decisions, this paper proposed a two-stage credit risk prediction framework TSCR, which did not modify any algorithm, and the existing algorithm was only a part of TSCR. In the first stage, the algorithm estimated the default probability of borrowers. The three-way decision used two thresholds α and β to split samples into POS, NEG, and BND areas according to their default probability. The threshold was automatically learned, and the optimization objective was represented by the quotient of the information gain of the three subsets and the overall decision cost. The optimal threshold pair was solved by the PSO algorithm. In the second stage, samples in the two determined regions POS and NEG were added to the existing data as supplementary information to form a new data set, and the algorithm model was retrained using this dataset to classify samples in the BND region.

We conducted experiments on a loan default dataset from a Chinese commercial bank, and three datasets with different data scales were constructed. Under the effect of information gain and decision cost, the Z value achieved the maximum in different threshold pairs. In the three-way decisions, the proportion of samples classified into the BND area under various algorithms was different. The highest ratio was more than 50% and the lowest was less than 1%. What is more, the classification performance of various algorithms under this framework had been improved. We also discussed the case that the BND region samples were judged by experts, and the classification effect was greatly improved in terms of FM and GM indicators.

The main contribution of the paper is to embed the three-way decisions into credit risk prediction and design a method of automatic threshold learning. In the learning of thresholds, we discussed the relationship between thresholds and the decision cost, and thresholds were set by considering the information gain and the decision cost. In future work, we will take the actual financial situation of different borrowers, such as loan amount and outstanding amount, into the decision loss, so that the threshold learning can be closer to the practical concerns of financial institutions. What is more, the defaulters are the minority, so credit risk prediction is a class-imbalanced problem. We will pay more attention to the impact of data imbalance on the application of three-way decisions in credit risk prediction.

ACKNOWLEDGMENTS

This work was funded by the National Natural Science Foundation of China (no. 72301093), the Natural Science Foundation of Hebei Province (no. G2023202004), and the Science and Technology Project of Hebei Education Department (no. QN2023121).

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Yusheng Li https://orcid.org/0000-0002-2921-2272

REFERENCES

Bao, W., Lianju, N., & Yue, K. (2019). Integration of unsupervised and supervised machine learning algorithms for credit risk assessment. *Expert Systems with Applications*, *128*, 301–315. https://doi.org/10.1016/j.eswa.2019.02.033

Deng, X., & Yao, Y. (2014). Decision-theoretic three-way approximations of fuzzy sets. *Information Sciences*, *279*, 702–715. https://doi.org/10.1016/j.ins.2014.04.022

Dumitrescu, E., Hué, S., Hurlin, C., & Tokpavi, S. (2022). Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects. *European Journal of Operational Research*, *297*(3), 1178–1192. https://doi.org/10.1016/j.ejor.2021.06.053

Dziwinski, P., & Bartczuk, L. (2020). A new hybrid particle swarm optimization and genetic algorithm method controlled by fuzzy logic. *IEEE Transactions on Fuzzy Systems*, *28*(6), 1140–1154. https://doi.org/10.1109/tfuzz.2019.2957263

Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. Paper presented at the MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science.

Fang, Y., Gao, C., & Yao, Y. (2020). Granularity-driven sequential three-way decisions: A cost-sensitive approach to classification. *Information Sciences*, *507*, 644–664. https://doi.org/10.1016/j.ins.2019.06.003

Fu, X., Ouyang, T., Chen, J., & Luo, X. (2020). Listening to the investors: A novel framework for online lending default prediction using deep learning neural networks. *Information Processing & Management*, *57*(4), 102236. https://doi.org/10.1016/j.ipm.2020.102236

Ge, R., Feng, J., Gu, B., & Zhang, P. (2017). Predicting and deterring default with social media information in peer-to-peer lending. *Journal of Management Information Systems*, *34*(2), 401–424. https://doi.org/10.1080/07421222.2017.1334472

Jia, X., Tang, Z., Liao, W., & Shang, L. (2014). On an optimization representation of decision-theoretic rough set model. *International Journal of Approximate Reasoning*, *55*(1, Part 2), 156–166. https://doi.org/10.1016/j.ijar.2013.02.010

Kou, G., Peng, Y., & Wang, G. (2014). Evaluation of clustering algorithms for financial risk analysis using MCDM methods. *Information Sciences*, *275*, 1–12. https://doi.org/10.1016/j.ins.2014.02.137

Lee, C.-S., Wang, M.-H., Wang, C.-S., Teytaud, O., Liu, J., Lin, S.-W., & Hung, P.-H. (2018). PSO-based fuzzy markup language for student learning performance evaluation and educational application. *IEEE Transactions on Fuzzy Systems*, *26*(5), 2618–2633. https://doi.org/10.1109/tfuzz.2018.2810814

Lee, J. W., Lee, W. K., & Sohn, S. Y. (2021). Graph convolutional network-based credit default prediction utilizing three types of virtual distances among borrowers. *Expert Systems with Applications*, *168*, 114411. https://doi.org/10.1016/j.eswa.2020.114411

Li, Y., & Shao, X. (2022). Thresholds learning of three-way decisions in pairwise crime linkage. *Applied Soft Computing*, *120*, 108638. https://doi.org/10.1016/j.asoc.2022.108638

Liang, K., & He, J. (2020). Analyzing credit risk among Chinese P2P-lending businesses by integrating text-related soft information. *Electronic Commerce Research and Applications*, *40*, 100947. https://doi.org/10.1016/j.elerap.2020.100947

Liu, J., Zhang, S., & Fan, H. (2022). A two-stage hybrid credit risk prediction model based on XGBoost and graph-based deep neural network. *Expert Systems with Applications*, *195*, 116624. https://doi.org/10.1016/j.eswa.2022.116624

Machado, M. R., & Karray, S. (2022). Assessing credit risk of commercial customers using hybrid machine learning algorithms. *Expert Systems with Applications*, *200*, 116889. https://doi.org/10.1016/j.eswa.2022.116889

Maldonado, S., Peters, G., & Weber, R. (2020). Credit scoring using three-way decisions with probabilistic rough sets. *Information Sciences*, *507*, 700–714. https://doi.org/10.1016/j.ins.2018.08.001

Oreski, S., Oreski, D., & Oreski, G. (2012). Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment. *Expert Systems with Applications*, *39*(16), 12605–12617. https://doi.org/10.1016/j.eswa.2012.05.023

Pan, R., Zhang, Z., Fan, Y., Cao, J., Lu, K., & Yang, T. (2016). Multi-objective optimization method for learning thresholds in a decision-theoretic rough set model. *International Journal of Approximate Reasoning*, *71*, 34–49. https://doi.org/10.1016/j.ijar.2016.01.002

Shen, F., Zhang, X., Wang, R., Lan, D., & Zhou, W. (2022). Sequential optimization three-way decision model with information gain for credit default risk evaluation. *International Journal of Forecasting*, *38*(3), 1116–1128. https://doi.org/10.1016/j.ijforecast.2021.12.011

Shi, Y., & Eberhart, R. C. (1999, 6-9 July 1999). Empirical study of particle swarm optimization. Paper presented at the Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406).

Song, Y., Wang, Y., Ye, X., Wang, D., Yin, Y., & Wang, Y. (2020). Multi-view ensemble learning based on distance-to-model and adaptive clustering for imbalanced credit risk assessment in P2P lending. *Information Sciences*, *525*, 182–204. https://doi.org/10.1016/j.ins.2020.03.027

Suo, M., Tao, L., Zhu, B., Chen, Y., Lu, C., & Ding, Y. (2020). Soft decision-making based on decision-theoretic rough set and Takagi-Sugeno fuzzy model with application to the autonomous fault diagnosis of satellite power system. *Aerospace Science and Technology*, *106*, 106108. https://doi.org/10.1016/j.ast.2020.106108

Suo, M., Tao, L., Zhu, B., Miao, X., Liang, Z., Ding, Y., Zhang, T., & Zhang, T. (2020). Single-parameter decision-theoretic rough set. *Information Sciences*, *539*, 49–80. https://doi.org/10.1016/j.ins.2020.05.124

Tariq, H. I., Sohail, A., Aslam, U., & Batcha, N. K. (2019). Loan default prediction model using sample, explore, modify, model, and assess (SEMMA). *Journal of Computational and Theoretical Nanoscience*, *16*(8), 3489–3503. https://doi.org/10.1166/jctn.2019.8313

Wang, L., Chen, Y., Jiang, H., & Yao, J. (2020). Imbalanced credit risk evaluation based on multiple sampling, multiple kernel fuzzy self-organizing map and local accuracy ensemble. *Applied Soft Computing*, *91*, 106262. https://doi.org/10.1016/j.asoc.2020.106262

Wang, T., Liu, R., & Qi, G. (2022). Multi-classification assessment of bank personal credit risk based on multi-source information fusion. *Expert Systems with Applications*, *191*, 116236. https://doi.org/10.1016/j.eswa.2021.116236

Xia, Y., He, L., Li, Y., Liu, N., & Ding, Y. (2019). Predicting loan default in peer-to-peer lending using narrative data. *Journal of Forecasting*, *39*(2), 260–280. https://doi.org/10.1002/for.2625

Xia, Y., Zhao, J., He, L., Li, Y., & Yang, X. (2021). Forecasting loss given default for peer-to-peer loans via heterogeneous stacking ensemble approach. *International Journal of Forecasting*, *37*(4), 1590–1613. https://doi.org/10.1016/j.ijforecast.2021.03.002

Xu, J., Zhang, Y., & Miao, D. (2020). Three-way confusion matrix for classification: A measure driven view. *Information Sciences*, *507*, 772–794. https://doi.org/10.1016/j.ins.2019.06.064

Xu, X., Zhou, C., & Wang, Z. (2009). Credit scoring algorithm based on link analysis ranking with support vector machine. *Expert Systems with Applications*, *36*(2), 2625–2632. https://doi.org/10.1016/j.eswa.2008.01.024

Yang, J., Wang, G., Zhang, Q., Chen, Y., & Xu, T. (2019). Optimal granularity selection based on cost-sensitive sequential three-way decisions with rough fuzzy sets. *Knowledge-Based Systems*, *163*, 131–144. https://doi.org/10.1016/j.knosys.2018.08.019

Yang, M., Lim, M. K., Qu, Y., Li, X., & Ni, D. (2023). Deep neural networks with L1 and L2 regularization for high dimensional corporate credit risk prediction. *Expert Systems with Applications*, *213*, 118873. https://doi.org/10.1016/j.eswa.2022.118873

Yao, Y. (2009). Three-Way Decision: An Interpretation of Rules in Rough Set Theory. Paper presented at the Rough Sets and Knowledge Technology, Berlin, Heidelberg.

Yao, Y. (2010). Three-way decisions with probabilistic rough sets. *Information Sciences*, *180*(3), 341–353. https://doi.org/10.1016/j.ins.2009.09.021

Yue, X. D., Chen, Y. F., Miao, D. Q., & Fujita, H. (2020). Fuzzy neighborhood covering for three-way classification. *Information Sciences*, *507*, 795–808. https://doi.org/10.1016/j.ins.2018.07.065

Zeitsch, P. J. (2019). A jump model for credit default swaps with hierarchical clustering. *Physica a: Statistical Mechanics and its Applications*, *524*, 737–775. https://doi.org/10.1016/j.physa.2019.04.255

Zhang, Q., Wang, J., Lu, A., Wang, S., & Ma, J. (2018). An improved SMO algorithm for financial credit risk assessment – Evidence from China's banking. *Neurocomputing*, *272*, 314–325. https://doi.org/10.1016/j.neucom.2017.07.002

Zhou, L., Si, Y.-W., & Fujita, H. (2017). Predicting the listing statuses of Chinese-listed companies using decision trees combined with an improved filter feature selection method. *Knowledge-Based Systems*, *128*, 93–101. https://doi.org/10.1016/j.knosys.2017.05.003

AUTHOR BIOGRAPHIES

Yusheng Li obtained his PhD at the Chinese Academy of Sciences in 2022. He is currently a lecturer at the School of Economics and Management, Hebei University of Technology. From 2021 to 2022, he was a visiting PhD student at the Department of Electrical and Computer Engineering, University of Alberta. His research interests include credit risk prediction and intelligent decisions.

Mengyi Sha is a postdoctoral researcher at the Department of Industrial Engineering, Tsinghua University. Before that, she was a postdoctoral researcher at the Department of Civil Engineering, McGill University. She holds a PhD degree in Management Science and Engineering from the University of Chinese Academy of Sciences. Her primary research interests include synergies between machine learning and operations research and data-driven optimization.

How to cite this article: Li, Y., & Sha, M. (2024). Two-stage credit risk prediction framework based on three-way decisions with automatic threshold learning. *Journal of Forecasting*, *43*(5), 1263–1277. https://doi.org/10.1002/for.3074