Journal of Information Systems and Informatics
Vol. 5, No. 2, June 2023 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v5i2.487 Published By DRPM-UBD

Machine Learning Approach for Credit Score Predictions

Tsholofelo Mokheleli[1], Tinofirei Museba[2]

[1,2]Department of Applied Information Systems, University of Johannesburg, Johannesburg, South Africa
Email: [1]217050248@student.uj.ac.za, [2]tmuseba@uj.ac.za

Abstract

This paper addresses the problem of managing the significant rise in requests for credit products that banking and financial institutions face. The aim is to propose an adaptive, dynamic heterogeneous ensemble credit model that integrates the XGBoost and Support Vector Machine models to improve the accuracy and reliability of risk assessment credit scoring models. The method employs machine learning techniques to recognise patterns and trends from past data to anticipate future occurrences. The proposed approach is compared with existing credit score models to validate its efficacy using five popular evaluation metrics, Accuracy, ROC AUC, Precision, Recall and F1_Score. The paper highlights credit scoring models' challenges, such as class imbalance, verification latency and concept drift. The results show that the proposed approach outperforms the existing models regarding the evaluation metrics, achieving a balance between predictive accuracy and computational cost. The conclusion emphasises the significance of the proposed approach for the banking and financial sector in developing robust and reliable credit scoring models to evaluate the creditworthiness of their clients.

Keywords: Credit Score, Machine learning, Class Imbalance, SMOTE, Ensemble, XGBoost, SVM

1. INTRODUCTION

Credit scoring models have emerged as effective and efficient tools for banks and other financial institutions to distinguish, recognise, and discriminate against potential default borrowers and mitigate credit risk. Given such a scenario, a credit scoring model's prediction, recognition, and discriminatory performance are important for financial institutions and banks to generate profits. Financial institutions use a credit score to determine a client's creditworthiness for a loan. Credit scores are generated by considering personal details such as historical track records on debt responsibilities, profiling, primary place of residence, earnings, job, demographic information, assets like vehicles and real estate, and census data. There has been a swift surge in the number of credit requests that financial institutions receive, and they have to assess the possible hazards associated with granting credit to their clients. The sooner financial institutions can ascertain whether or not to provide credit to their clients, the more advantageous it is. Credit scores are utilised by lenders, retailers, car dealerships, and real estate agents to appraise whether a client is eligible for a loan, credit card, automobile, or a new residence. Additionally, they determine the interest rate and credit limit that are applicable.

Credit scoring is useful for managing credit risk and minimising information asymmetry [1] [2]. Its purpose is to produce a score that can differentiate loan applicants into two categories: those who are creditworthy and likely to repay their loans and those who are risky and unlikely. This score is linked to the anticipated likelihood of default and is transformed into a classification task [3]. The creation of a robust, efficient, and adaptable credit scoring model has a significant impact on the profitability of financial institutions [4]. Every credit risk scoring model must comply with stringent regulations, and any violation may result in significant regulatory costs. Therefore, creating credit scoring models that are adaptable, efficient, and robust in accurately predicting loan defaults is crucial. Before the advent of machine learning, statistical models were used for credit scoring. Nevertheless, statistical methods usually rely on strong assumptions such as linear separability and normal distribution of data [5]. These assumptions can restrict statistical methods' effectiveness when applied to large datasets or when they are violated.

Credit scoring is generally computed using different mathematical tools that estimate the probability of default (PD) of the party receiving the loan [6]. While this approach can provide valuable insights into a customer's creditworthiness, the traditional data analysis methods and manual credit scoring methods can be slow and resource intensive. As a result, banks are increasingly turning to machine learning and other automated techniques to speed up the credit evaluation process and make more accurate predictions [7]. These technologies can analyse large volumes of data and identify patterns and trends that may be difficult for humans to detect, enabling banks to make faster, more informed lending decisions.

The study aims to develop a model that delivers precise results even when the data is unbalanced, and customer variables are subject to change over time. To address imbalanced data, oversampling is used, which adjusts unequal data classes to generate balanced datasets. The model's effectiveness was evaluated using various real-life credit score datasets. The study also explored imbalance classification, which involves building prediction models based on classification datasets with a notable class imbalance. Dealing with unbalanced datasets can be challenging as many machine learning techniques tend to neglect the minority class. This can lead to poor performance, even though accurately identifying the minority class is often the most important aspect [8]. In this study, various datasets containing different classes were used to tackle the issue of imbalanced data, with some datasets having more positive samples and others having more negative samples. To address the class imbalance problem, the Synthetic Minority Oversampling Technique (SMOTE) was applied to oversample the minority class. This technique involves randomly increasing the number of minority class samples by replicating them to balance the class distribution. After SMOTE, Principal Component Analysis (PCA) was conducted to identify the key features that significantly impact the results. PCA is an unsupervised learning method that reduces dimensionality in machine learning. It uses an orthogonal transformation statistical technique to transform the observations of correlated variables into a set of linearly uncorrelated data [9]. This study's primary achievement is creating an adaptive, dynamic, and novel heterogeneous ensemble credit scoring model. Our proposed model differs from existing models in that it considers changes that occur with customer variables over time and applies the dynamic ensemble selection to incorporate both accuracy and diversity.

The data utilised in the experiments were obtained from the UCI Machine Learning Repository [10]. These datasets are widely used in related research, making comparing predictions comprehensively with existing studies feasible. Some of the datasets were obtained from the Kaggle and UCI repositories. They are accessible to the public and can be downloaded free of charge.

1.2. Related Work

Banks and most financial institutions use a quantitative model for credit scoring to distinguish between creditworthy and risky customers. Given the increasing complexity of credit scoring, several approaches to designing efficient and robust scoring models have been proposed. Recently, machine learning ensembles have gained prominence over statistical models due to their recognition and adaptation abilities in developing robust and assertive credit scoring models. Credit scoring models utilise information from loan applications and customer details to accurately predict the likelihood of loan default. Credit scoring is a crucial aspect of the credit risk management system for the majority of financial institutions, aiding in the prediction of a surge in loan applications, and a number of contemporary approaches based on ensembles of machine learning approaches to establish quantitative credit scoring models to distinguish between two categories of applications namely: creditworthy applicants and non-creditworthy applicants. Due to its strong interpretability of results, Zhang et al. [11] applied the logistic regression model to design a novel ensemble called the Balancing and Weighting Effect (BWE). The major drawback of the logistic Balancing and Weighting Effects model is that the balancing operation of training samples enhances the recognition ability of default samples at the expense of the recognition ability of non-default samples. BWE requires o be integrated with other credit scoring models and learning algorithms to increase diversity and further improve the recognition ability of the BWE.

To accurately handle the class imbalance problem that is inherent in credit scoring, as the misclassification of the minority is often costly, Johah Mushava and Michael Murray [12] suggested utilising XGBoost, a dependable and effective classification technique and included the quantile function of the Generalized Extreme Value (GEV) distribution as a link function to improve the identification of infrequent cases. While XGBoost-based methods are intricate and offer superior outcomes compared to simple imputation methods, these techniques lessen the comprehensibility of the scoring outcomes. In another research paper, the same authors, Johah Mushava and Michael Murray [13] investigated the predictive power of the most popular classification technique currently used for credit scoring, with special attention to predicting a client to pay given different intervals of days in arrears. The approach only works well for a fixed window of 3 to 12 months, but most clients can go beyond even five years with no payment made. The approach does not consider the occurrence of variable drifts. Developing a reliable and confident credit scoring model takes considerable time, usually between 3 to 18 months. Therefore, it is not uncommon for financial institutions and credit scoring models to remain unchanged for several years.

The credit scoring task should be considered an ephemeral scenario since variables can drift over time. Yiqiong Wu [14] proposed a credit scoring framework that focuses on uncertainty and incorporates multi-objective feature selection to handle credit classification under uncertain conditions. The multi-objective optimisation problem is addressed using a modified evolutionary algorithm and a binary multi-objective particle swarm optimisation. One of the drawbacks of this approach is that it uses a simple dummy method to encode categorical variables. The experimental results show that the credit scoring model with better AUC and AUCC values may only sometimes yield satisfactory FPR or FNR values. The method for determining the cut-off point could be more effective. Hongliang He [15] introduced a new ensemble model to tackle the problem of class imbalance in credit datasets. This model can adjust the imbalance ratios to enhance recognition performance. The proposed approach extends the supervised under-sampling approach called BalanceCascade to create adjustable datasets to estimate data imbalance ratios. The proposed method comprises three stages and employs the PSO algorithm to optimise parameters. It adopts a stacking approach to combine RF and XGBoost as base classifiers to form an ensemble. Despite the recommendations for improving the handling of imbalanced data, the approach still has limitations. For instance, it does not consider the impact of redundant samples from positive classes and the performance of an ensemble model with more than three base classifiers. Nevertheless, the models' diversity is key to any ensemble classifier's success.

Wanan Liu [13] proposed two tree-based augmented GBDTS for credit scoring [16]to harness the power of tree-based algorithms for credit scoring models. Diversity is introduced via a stepwise feature augmentation mechanism. The proposed approach was evaluated on four large-scale credit scoring datasets along with several benchmark models, and the performance comparison demonstrated that the proposed approach is effective. The proposed approach needs to integrate tree-based stepwise feature augmentation with XGBoost making the performance poorly balanced, more complex, and difficult to interpret the results. A credit scoring model that incorporates the bagging algorithm with the stacking method was proposed by Yufei Xia et al. [17]. The Bstacking model involves four base learners trained in bagging samples. However, complex models like Bstacking may raise privacy concerns and regulatory actions. In addition, interpretability should be highlighted to balance a real-world credit scoring model's accuracy, complexity, and interpretability. Credit scoring involves working with large amounts of data, which makes it difficult to perform resampling during model training. As a result, methods such as bagging and boosting, which involve resampling the training data, are typically not used in credit scoring. In another effort to create a credit score model capable of accurately distinguishing loan applicants, Yufei Xia et al. [17] proposed a credit scoring model called the overfitting-cautious heterogeneous ensemble model (OCHE) is a tree-based heterogeneous ensemble model designed to avoid overfitting. This model uses a dynamic ensemble selection strategy and advanced tree-based classifiers as base models. The approach also considers overfitting in the ensemble selection stage. The proposed model was compared with benchmark models on five publicly available real-world datasets. It outperformed most individual and homogeneous ensemble models regarding predictor accuracy, as measured by four metrics.

Using a powerful base such as XGBoost and CatBoost, which are complex but generate better results, may also lead to the deterioration of the interpretability of the scoring results. This paper proposes the Adaptive Dynamic Heterogeneous Ensemble (ADHE) that explores the dynamic ensemble selection to formulate an ensemble of accurate and diverse models derived from two base learners. To detect and adapt to changes in the behaviour of applicants, models are updated regularly. To tackle the class imbalance issue, the Synthetic Minority Oversampling Technique (SMOTE) is utilised. The XGBoost algorithm is used for feature processing.

2. METHODS

Ensemble learning combines the prediction outputs of different classifiers to generate better generalisation than applying a single algorithm. This section proposes the Adaptive and Dynamic Heterogeneous Ensemble (ADHE) for credit scoring. Adaptive and Dynamic Heterogeneous Ensemble is a machine learning technique that combines multiple models to improve prediction accuracy and robustness. It involves creating an ensemble of diverse models that complement each other's strengths and weaknesses. The adaptive and dynamic aspect refers to the ability to adjust the ensemble in response to data and environment changes. In the credit scoring context, the Adaptive and Dynamic Heterogeneous Ensemble approach integrates XGBoost and Support Vector Machine models to create a more accurate and reliable credit scoring model. XGBoost is a gradient-boosting algorithm that can handle large and complex datasets while Support Vector Machines effectively handle high-dimensional data. By combining these models, the ensemble is better equipped to handle the challenges of credit scoring, such as class imbalance, verification latency, and concept drift. The experimental setup is specified from aspects such as Dynamic Ensemble Selection and pool generation, base learners, data pre-processing, hyperparameter tuning, evaluation metrics and credit datasets.

2.1. Dynamic Classifier Selection and Pool Generation

Given a training dataset, D_{train} = {x,y}, where x is an MXN dimensional feature matrix and y ∈ {0,1}N indicates the label. A value of 1 in y represents a default application, whereas 0 is an indication of creditworthiness. The dataset D_{train} generates an initial pool of classifiers from the two base learning algorithms: XGBoost and Support Vector Machines. Our proposed approach employs a heterogeneous ensemble architecture. After the initial pool is generated, classifier ensembles are subsequently selected. The approach for selection is dynamic, and the classifiers considered competent are chosen using a fitness function specific to different groups of test samples. This makes the ensemble classifier be created dynamically [18]. For an ensemble to accurately distinguish applicants, base models in ensemble learning must be diverse and accurate. This study selects classifiers based on their accuracy on the validation set and their diversity to handle the incremental learning that accounts for the changing customer behaviour over time. The credit scoring task is transitory because various variables might alter over time. Therefore, the study uses data stream mining techniques designed for incremental learning and to detect and adjust to changes in the data distribution. To select classifiers from the pool that are accurate and diverse, we employ the algorithm called Selection by Accuracy and Diversity (SAD) [19], which is as follows:

1) Train a set of different classifiers
2) Measure the accuracy of each classifier on a validation set.
3) Choose the top-performing classifiers based on accuracy.
4) Measure the diversity between the chosen classifiers and the remaining ones.
5) Select additional classifiers with high diversity and add them to the ensemble until the desired size is reached.
6) Combine the classifiers in the ensemble.
7) Evaluate the performance of ensemble learning.

The Q Statistic [20] diversity measure is used as a diversity measure in this study due to its simplicity and ease of interpretation.

2.2. Base Learners

In simple terms, the success of a classifier ensemble relies on the diversity of performance of its base classifiers [21]. Our approach uses two base learning algorithms, XGBoost and Support Vector Machines (SVM), to introduce diversity. In addition to the introduction of diversity, the two base learners have the potential to strike a good balance between accuracy and efficiency. Support Vector Machines have demonstrated tremendous capability for regression and classification problems in static and dynamic domains. They have been extensively used to address the curse of dimensionality for most classification problems. For classification tasks, SVM can identify a hyperplane that effectively separates linear data into two classes while maximising the distance between the training instances. If the data is non-linear, the SVM kernel function maps it to a higher dimensional space. In such cases, SVM looks for an optimal hyperplane that can separate the two data classes in the high-dimensional feature space. Support Vector Machine has two hyperparameters, the cost parameter c and the RBF kernel parameter γ.. The cost parameter manages both the misclassification and the complexity level. The RBF kernel parameter regulates the impact of an individual training sample on the hyperplane.

Chen and Guestrain [22] developed eXtreme Gradient Boosting (XGBoost) to address classification problems encountered in real-world scenarios. XGBoost can reduce model variances by incorporating regularisation into the loss function and using a weighted quantile sketch for tree learning to handle sparse data. XGBoost surpasses many other machine learning algorithms in speed and accuracy because of these techniques and weights. It uses Taylor's expansion to approximate the loss function quickly.

The XGBoost learning algorithm has quite a number of hyperparameters. The number of estimators' hyperparameters controls the number of iterations in XGBoost. The maximum depth hyperparameter determines the maximum depth of a single base learner, while the subsampling rate hyperparameter specifies the fraction of samples used to train one base learner. The learning rate hyperparameter reduces the contribution of each base learner. The column sampling rate hyperparameter determines the fraction of features used for training a single base learner. Finally, the gamma hyperparameter determines the minimum loss reduction necessary to create a new partition.

2.3. Parameter Optimisation

The base learners employed in the study, Support Vector Machines and XGBoost, are associated with several parameters that can substantially impact the prediction performance of the credit card fraud detection system. The parameters must be optimised for the fraud detection system to perform optimally, and several optimisation algorithms exist. Most existing optimisation techniques suffer from the curse of dimensionality. The computational cost involved tends to increase dramatically with the number of hyperparameters or as the search space is extended. The tuning of hyperparameters for most applications is subjective and relies on empirical judgement and trial and error approaches. To overcome the drawbacks of existing optimisation algorithms, this study employs an adaptive heterogeneous Particle Swarm Optimizer to appropriately optimise and generate an optimal subset of accurate parameters and improve the efficacy of XGBoost and Support Vector Machines for the classification problem. Kennedy and Eberhart [23] created the Particle Swarm Optimization (PSO) algorithm, a popular heuristic algorithm, and an evolutionary computational technique. The PSO algorithm is a population-based, iterative, global, and stochastic optimisation technique. It takes inspiration from the social behaviour of birds flocking or fish schooling to conduct an intelligent search for the best possible solution [24]. PSO is a heuristic optimisation algorithm that does not need gradients as it is not based on differentiability. This makes it useful for solving problems that have non-convex or discontinuous functions. In the current research, the swarm's particles were instantiated individually to introduce diversity within the swarm. This allows for different search behaviours among the particles as they can randomly choose velocity and position update rules from a pool of possible behaviours. Combining exploratory and exploitative particles allows the algorithm to balance exploration and exploitation, preventing premature convergence and allowing for a better solution space search.

2.4. Data Pre-processing

The datasets utilised in this research are processed through standardisation, scaling to a range of 0 to 1, and the approximation of missing values. The class imbalance issue in the credit card fraud datasets is also addressed. The mean is removed and scaled to unit variance to standardise numeric features. The data is scaled using the 0-1 normalisation method. If x is a given feature, then the normalised feature can be calculated as follows:

$$X^{'} = \frac{x\text{-}min\boxed{f0}(x)}{max(x).min\boxed{f0}(x)} \tag{1}$$

where x' expresses the standardised value.

Normalising features significantly enhances the precision of classifiers, particularly those that rely on distance or edge computations, making the model more confident and precise. Credit card fraud data is associated with class imbalance. Detecting credit card fraud is challenging due to the highly skewed distribution of credit card transaction data, where the proportion of legitimate transactions (majority class) significantly outweighs the proportion of fraudulent transactions (minority class) in the real world. Credit card fraud data is also associated with missing values. XGBoost includes a technique called sparsity segmentation that can accurately estimate missing values. Standardisation is applied to reduce the impact of outliers, and centralisation is used to address extreme values. The class imbalance problem is addressed using Synthetic Minority Oversampling Technique (SMOTE), a resampling technique. In a study by Chawla N. V [25], the recognition performance of classifiers for the minority class was improved using Synthetic Minority Oversampling Technique (SMOTE). SMOTE generates artificial cases similar to the observed ones, oversampling the minority class.

Additionally, a metric called degOver is utilised to address class imbalance with overlap, which considers both the imbalance ratio and the dataset structure [26]. Dynamic Ensemble Selection (DES) is applied to handle different drifting concepts. To handle verification latency, we employ integrated Fraud Detection (FD) [27]. Smooth Clustering based Boosting (SCBoost) is a fraud detection method with noise-resistant boosting. It is combined with k-Shortest Distance Ratio (k-SDR), which helps to effectively use the labelled dataset and address issues caused by class imbalance. K-SDR's primary function is to classify an instance based on the ratio of its average distance to the k nearest instances in the positive class, preventing any interference caused by a class imbalance in the labelled dataset.

2.5. Feature Selection

Feature selection is carried out using XGBoost. It computes feature importance scores by measuring the average reduction in objective function value achieved using a particular variable for splitting. This evaluation is carried out immediately when variables are selected for splitting. During the tree-building process, variables with higher scores are considered more important. This study employs XGBoost as a joint base learner with Support Vector Machine, and the suggestions proposed by Xia [28] are followed to implement scores derived from feature importance as a guideline in a sequential forward search (SFS) feature selection algorithm. SFS places the relevant features into the subset and iteratively adds the features that remain and have the highest scores, thus generating a series of candidate feature subsets. Only the feature subset that maximises the cross-validated accuracy is selected as the optimal feature set suitable for training the model in the subsequent steps.

2.6. Performance Metrics

The study presented in this paper is modelled as a machine learning binary classification task. We selected five popular evaluation metrics to comprehensively perform our proposed approach, ADHE and benchmarks. The performance evaluation metrics were selected due to their popularity in the literature on existing credit scoring. The performance metrics include accuracy.

The accuracy obtained from the test data is used as the main performance metric. Furthermore, we compute each model's Precision, Recall, F1_Score and Area Under the Curve (AUC). The Area Under the Curve provides a proper assessment of the classification quality of each model. The AUC metric provides a measure of the effectiveness of a classifier for a given task. The value of AUC is within the interval 0 to 1, and an efficient classifier is identified with an AUC value almost close to 1. The accuracy metric is determined by dividing the total number of accurate predictions by the overall number of forecasts made.

On the other hand, precision refers to the ratio of the total number of accurate predictions made to the total number of correct predictions made. A recall metric measures the proportion of correct predictions of positive class values in the test dataset. Finally, the F1 score is a measure that represents the equilibrium between accuracy and recall. The performance metrics can be expressed mathematically as follows:

$$Accuracy = \frac{TN+PP}{TP+TN} \tag{2}$$

$$Recall = \frac{TP}{FN+TP} \tag{3}$$

$$Precision = \frac{TP}{FP+TP} \tag{4}$$

$$F1\ score = 2\frac{PR.RC}{PR+RC} \tag{5}$$

3. RESULTS AND DISCUSSION

3.1. Experimental Design

3.1.1. Data Description

Credit scoring in the era of big data has its challenges. Credit data is big and often nonstationary. Data is constantly evolving as customers' behaviour changes. Real-time processing systems have significant business value because they can react instantly. Machine learning models, in many cases, are built on outdated data that no longer accurately represents the distribution of new data. The main difficulty is promptly detecting and adapting to changes in concept drift and successfully managing model transitions during these changes. Credit data is typically non-linear and has many points, creating a dense cloud that makes it challenging to observe relationships and determine linearity. Along with concept drift and nonlinearity, credit scoring data also has a class imbalance issue where some classes have many samples and others only have a few. The overall performance of a machine learning algorithm can be adversely affected when large datasets contain data from classes with different probabilities of occurrence.

Five real-world credit datasets are employed to validate our proposed model's efficacy. Among the five datasets, the three most popular ones, Australian, Japanese, and German, are sourced from the UCI Machine Learning Repository [10]. The selected datasets are frequently used in related literature, enabling us to perform a feasible comparison with other state-of-the-art studies. The three datasets need to be larger to perform a detailed analysis of the behaviour of our proposed model. We added two other large credit datasets for a detailed and accurate comparison with existing studies. The Peer to Peer (P2P) consists of quite a number of instances regarding consumer lending, and a dataset from PPDai is employed. The PPDai [15] consists of instances of transaction records sourced from an advanced P2P lending platform in China. From the Kaggle community, we sourced the 'Give Me Some Cash' (GMSC) dataset. The Australian and Japanese datasets comprise 690 instances, where 307 samples are good ones, and 383 samples are default ones. The only notable difference between the two datasets is the number of features. The Australian dataset comprises eight numerical and six categorical features, whereas the Japanese dataset comprises five numerical features and ten categorical features. The German dataset consists of 1 000 samples, of which 700 are paid up, and 300 are default. The categorical features for the German dataset are 13, and the numerical features are 7. The PPDai dataset comprises 55 596 instances, of which 48 413 are considered good, and 7 183 are default. The dataset also includes 29 features, 22 of which are numeric and the remaining 7 categorical. The GMSC dataset comprises 150 000 samples 139 974 are fully paid, and the remaining 10 126 instances represent the default ones. A summary of the datasets used is provided as follows.

Table 1. A summary of the credit datasets

Name	Abbreviations	Samples	features	Good/bad	Source
Australian	Australian	690	14	307/383	UCI: Machine Learning Repository [10]
Japanese	Japanese	690	15	307/383	UCI: Machine Learning Repository [29]
German	German	1 000	24	700/300	UCI: Machine Learning Repository [30]
PPDai	PPDai	55 596	29	484413/7183	[15]
Kaggle GMSC	GMSC	150 000	10	139974/10026	http://www.kaggle.com/GiveMeSomeCash

3.1.2. Benchmark Models

Various classifiers and benchmarks are used to perform a comparative performance of our proposed approach. Five individual classification techniques and five ensemble models are employed. The five individual classification techniques used are KNN, RF, XGBoost, LR and SVM. They have been selected as individual base learners as they are most commonly employed as benchmark models in the credit scoring domain. Since the prediction performance of ensemble models is demonstrated in the literature, we use the Overfitting-cautious heterogeneous ensembles model (OCHE) [31], bagging algorithm with stacking method (BStack) [17], a group method of data handling (GMDH) based sensitive semi-supervised selection ensemble (GCSSE) model [32], the Generalised Shapley Choquet Integral (GSCI) [33] and the Adaptive Particle Swarm Optimization [34].

3.2. Experimental Results

A comprehensive comparison of the prediction performance of our proposed approach with the selected benchmark models. The first experiments compare our proposed prediction performance against individual base learners and homogeneous ensemble models. The proposed and benchmark models are validated on five credit score datasets across five evaluation metrics. The empirical experiments are conducted in Python 2.7 on a PC with 3.6 GHz, Intel i7 CPU, 8GB RAM and Microsoft Windows 10 Operating System. Table 2 provides the average prediction performances of individual classifiers and homogeneous ensembles against our ADHE approach on seven evaluation measures.

Table 2. Performance of individual classifiers across datasets

Dataset	Model	Accuracy %	Precision (0-1)	Recall (0:1)	F1_Score	G-Mean	ROU-AUC	Kappa
Australian	KNN	69.0	0.81: 0.49	0.73: 0.60	0.77: 0.54	0.663	0.666	0.31
	RF	73.7	0.80: 0.57	0.83: 0.53	0.81: 0.55	0.661	0.679	0.36
	XGBoost	71.2	0.79: 0.52	0.80: 0.51	0.79: 0.51	0.636	0.652	0.31
	LR	74.2	0.83: 0.53	0.75: 0.65	0.79: 0.58	0.696	0.697	0.37
	SVM	73.6	0.86: 0.55	0.75: 0.71	0.80: 0.62	0.730	0.730	0.43
	DAHE	76.4	0.82: 0.60	0.83: 0.58	0.82: 0.59	0.694	0.705	0.41
Japanese	KNN	87.0	0.89: 0.84	0.91: 0.80	0.90: 0.82	0.854	0.856	0.31
	RF	87.9	0.88: 0.87	0.93: 0.78	0.91: 0.82	0.855	0.858	0.36
	XGBoost	86.2	0.90: 0.81	0.89: 0.82	0.89: 0.82	0.854	0.854	0.31
	LR	85.5	0.89: 0.79	0.87:0.82	0.88: 0.81	0.848	0.849	0.37
	SVM	84.1	0.92: 0.74	0.82: 0.88	0.87: 0.80	0.849	0.849	0.43

Dataset	Model	Accuracy %	Precision (0-1)	Recall (0:1)	F1_Score	G-Mean	ROU-AUC	Kappa
	DAHE	87.7	0.89: 0.85	0.92: 0.80	0.90: 0.83	0.860	0.862	0.41
German	KNN	72.8	0.85: 0.19	0.82: 0.23	0.84: 0.21	0.431	0.524	0.05
	RF	79.6	0.85: 0.25	0.92: 0.14	0.88: 0.18	0.364	0.532	0.08
	XGBoost	82.5	0.85: 0.30	0.97: 0.08	0.90: 0.12	0.270	0.521	0.06
	LR	62.8	0.90: 0.24	0.63: 0.63	0.74: 0.35	0.629	0.629	0.16
	SVM	61.5	0.89: 0.23	0.63: 0.60	0.74: 0.34	0.615	0.615	0.14
	DAHE	79.6	0.86: 0.29	0.91: 0.20	0.88: 0.24	0.426	0.554	0.12
PDDai	KNN	72.8	0.85: 0.19	0.82: 0.23	0.84: 0.21	0.431	0.524	0.05
	RF	77.6	0.85: 0.25	0.92: 0.14	0.88: 0.18	0.364	0.532	0.08
	XGBoost	80.5	0.83: 0.30	0.97: 0.08	0.90: 0.12	0.270	0.521	0.06
	LR	62.8	0.93: 0.24	0.64: 0.63	0.74: 0.35	0.629	0.629	0.16
	SVM	67.5	0.87: 0.23	0.67: 0.60	0.76: 0.34	0.665	0.645	0.24
	DAHE	82.3	0.83: 0.29	0.93: 0.20	0.89: 0.24	0.446	0.564	0.32
GMSC	KNN	74.6	0.85: 0.19	0.82: 0.23	0.84: 0.21	0.431	0.524	0.05
	RF	78.6	0.85: 0.25	0.92: 0.14	0.88: 0.18	0.364	0.532	0.08
	XGBoost	79.5	0.85: 0.30	0.97: 0.08	0.90: 0.12	0.270	0.521	0.06
	LR	64.8	0.90: 0.24	0.63: 0.63	0.74: 0.35	0.629	0.629	0.14
	SVM	63.5	0.89: 0.23	0.63: 0.60	0.74: 0.34	0.615	0.615	0.15
	DAHE	81.7	0.76: 0.29	0.89: 0.20	0.87: 0.24	0.466	0.554	0.13

The section compares results from our proposed ADHE approach with the individual classifiers and homogeneous ensembles on five credit-scoring datasets across seven performance metrics. Table 4 presents the Accuracy scores of the ADHE model and other homogeneous ensemble models used in previous studies. From the results, it can be inferred that the ADHE model generally outperforms the homogeneous ensembles. Additionally, the heterogeneous ensemble is constructed through a feature selection process, which further improves the proposed model's performance. The prediction performance of the proposed ADHE is better in general than the homogeneous ensembles, reflecting the ADHE approach's effectiveness. The experimental results for the ADHE heterogeneous ensemble model for all seven-evaluation metrics are the best for the datasets from Australian, Japanese, German, PPDai and the GMSC. The prediction performance of single classifiers and homogeneous ensembles could be more consistent and stable on different datasets.

3.2.1. Comparison of the ADHE and Ensemble Benchmarks

The performance of ADHE is compared to the other five state-of-the-art ensemble models. The results are shown in Table 4. The tables reveal the findings of all the experiments conducted on the five datasets. ADHE performs the best overall on all evaluation metrics. The prediction performance of ADHE is enhanced further by the selection of competent classifiers that are diverse, making it able to adapt to changes in the underlying distribution of the data.

Since the prediction performance of ensemble models is demonstrated in the literature, we employ the Overfitting-cautious heterogeneous ensembles model (OCHE) [31], bagging algorithm with stacking method (BStack) [17], a group method of data handling (GMDH) based sensitive semi-supervised selection ensemble (GCSSE) model (Xu Zhou, 2020), the Generalised Shapley Choquet Integral (GSCI) [33] and the Adaptive Particle Swarm Optimization (APSO-XGBoost, 2021) [34].

Table 3. Performance results of ensemble models

Dataset	Model	Accuracy	Precision	Recall	F1-Score	G-Mean	ROU-AUC	Kappa
Australian	OCHE	78.69	0.85: 0.23	0.82: 0.32	0.84: 0.29	0.428	0.531	0.08
	Bstacking	79.63	0.85: 0.29	0.92: 0.19	0.88: 0.17	0.359	0.546	0.05
	GCSSE	76.3	0.85: 0.43	0.97: 0.12	0.90: 0.13	0.293	0.537	0.07
	GSCI	66.7	0.90: 0.27	0.63: 0.67	0.74: 0.39	0.653	0.633	0.17
	APSO-XG	69.4	0.89: 0.28	0.63: 0.64	0.74: 0.31	0.628	0.629	0.14
	DAHE	89.6	0.79: 0.31	0.86: 0.23	0.89: 0.29	0.479	0.574	0.16
Japanese	OCHE	75.7	0.83: 0.21	0.81: 0.24	0.81: 0.23	0.443	0.532	0.07
	Bstacking	79.5	0.82: 0.27	0.92: 0.26	0.86: 0.19	0.368	0.548	0.06
	GCSSE	77.6	0.83: 0.32	0.97: 0.13	0.89: 0.13	0.273	0.523	0.08
	GSCI	67.9	0.88: 0.27	0.64: 0.69	0.73: 0.36	0.624	0.621	0.16
	APSO-XG	68.3	0.83: 0.26	0.69: 0.61	0.74: 0.34	0.621	0.624	0.14
	DAHE	93.4	0.72: 0.31	0.83: 0.27	0.81: 0.29	0.476	0.564	0.12
German	OCHE	76.5	0.83: 0.23	0.84: 0.26	0.81: 0.23	0.453	0.532	0.05
	Bstacking	75.8	0.83: 0.28	0.89: 0.19	0.83: 0.19	0.357	0.548	0.06

Dataset	Model	Accuracy	Precision	Recall	F1-Score	G-Mean	ROU-AUC	Kappa
	GCSSE	77.4	0.82: 0.38	0.91: 0.07	0.89: 0.13	0.272	0.532	0.08
	APSO-XG	67.6	0.83: 0.28	0.65: 0.68	0.76: 0.38	0.631	0.631	0.16
	DAHE	65.8	0.85: 0.26	0.62: 0.62	0.77: 0.38	0.627	0.628	0.14
PDDai	OCHE	77.6	0.83: 0.19	0.79: 0.23	0.82: 0.21	0.426	0.538	0.06
	Bstacking	73.6	0.81: 0.25	0.87: 0.14	0.86: 0.18	0.372	0.543	0.07
	GCSSE	77.5	0.82: 0.30	0.83: 0.08	0.87: 0.12	0.268	0.521	0.05
	APSO-XG	67.8	0.86: 0.24	0.67: 0.63	0.76: 0.35	0.624	0.628	0.13
	DAHE	62.5	0.82: 0.23	0.69: 0.60	0.72: 0.34	0.623	0.621	0.14
GMSC	OCHE	73.6	0.87: 0.19	0.79: 0.23	0.78: 0.21	0.454	0.538	0.06
	Bstacking	76.6	0.84: 0.25	0.89: 0.14	0.83: 0.18	0.373	0.563	0.07
	GCSSE	78.5	0.83: 0.30	0.86: 0.08	0.89: 0.12	0.281	0.546	0.08
	APSO-XG	69.8	0.82: 0.24	0.64: 0.63	0.71: 0.35	0.633	0.634	0.15
	DAHE	67.5	0.89: 0.23	0.62: 0.60	0.72: 0.34	0.628	0.623	0.13

The prediction performance of the created ensemble is evaluated by comparing it with the individual classifiers. The prediction results for all five datasets are presented, and Table 2 displays the prediction performance of both the individual models and the ensemble, using various indicators. The prediction performance of ADHE and other benchmark models is relatively good. This is hugely attributed to the simultaneous consideration of accuracy and diversity of both learners in the combination stage. Tables 3 reveal important findings of the behaviour of ADHE in handling changes and class imbalance. Firstly, ADHE outperforms the rest of the benchmark ensemble models and achieves first place among the evaluation metrics for most datasets. Secondly, other ensemble-based approaches achieve good performance, demonstrating the superiority of heterogeneous ensemble methods in credit scoring. OCHE and Bstacking perform well as they show acceptable results across the five datasets, which partially explains why they are often selected as benchmarks for most new credit scoring approaches. For several datasets, benchmark models show acceptable results. As shown on other performance metrics, the prediction performance of benchmarks exhibits different behaviours, which inversely necessitates evaluating benchmark models from various aspects such as label, probability, and discriminatory capability. The results in the Table 3 demonstrate the advantages of heterogeneous ensemble approaches in credit scoring. Ensemble methods built using accuracy and diversity provide promising prediction performance, especially in credit scoring.

3.2.2. Comparison of Computational Cost

An effective and robust credit scoring model has to be computationally efficient. In addition, to be computationally efficient, a credit scoring model must provide quick and accurate responses to prospective loan applicants. The credit scoring model must consider that the variables differ for each applicant since the variables drift with time, and the changes to the training model must be consistent with the frequent updates of the credit scoring model. The application of the XGBoost as one of the base learners supports Graphics Processing Unit (GPU) to perform highly parallel independent calculations, significantly reducing computational time. This section compares the computational cost of benchmark models and our proposed model called ADHE. To accurately measure the computational cost, we implement a single training time.

Furthermore, it is calculated as the whole training time of a single cross-validation. Table 4 shows that the training time of ADHE given the GPU support end up as 1.96, 2.78., 4.56, 9.86 and 26.28 for Australian, Japanese, German, PPDai and GMSC. Table 4 also shows the single training time for the benchmark models. Comparing our proposed approach and the benchmark models regarding computational cost revealed a trade-off between computational cost and model prediction performance.

Table 4. Comparison of the computational cost of benchmark models

Benchmark	Australian	Japanese	German	PPDai	GMSC
OCHE	1.85	2.15	3.76	8.93	9.78
BStacking	1.78	1.84	2.13	6.43	7.85
GCSSE	2.84	2.93	2.34	7.65	9.04
GSCI	1.12	1.49	1.63	4.36	6.73
APSO-XGBoost	1.24	1.29	1.56	3.86	7.41
ADHE	2.78	3.46	5.84	11.82	31.65
ADHE(GPU)	1.96	2.78	4.56	9.86	26.28

3.2.3. Statistical Significance Tests

For classification problems, each performance metric has its own merits and demerits. To evaluate our proposed model against benchmark models, we used a non-parametric significance test instead of a parametric test because when comparing credit-scoring models, the assumptions of parametric tests are frequently unmet. This test can establish the statistical significance between the models and assess the performance differences. In their study, Lessman et al. [35] utilise non-parametric tests to compare classification models, as parametric tests [36] are often not suitable for such comparisons due to the assumptions they make.

They employ the Friedman, non-parametric test that ranks the models to assess their differences. It calculates a statistic based on the following formula:

$$\chi_F^2 = \frac{12D}{K(K+1)} \left[\sum_{k=1}^{K} AvR_j^2 - \frac{k(k+1)^2}{4} \right] \tag{6}$$

where

$$AvR_j^2 = \frac{1}{D} \sum_{i=1}^{D} r_i^j \tag{7}$$

D and K represent the number of datasets and classifiers, respectively, and r_i^j denotes the averaged rank of classifier j on dataset i. To calculate the average rank for each classifier, we use the corresponding rank among the evaluation metrics over datasets without losing any generality. Suppose the Friedman test null hypothesis is false, indicating a significant difference in the average ranks of the models for a specific evaluation measure. In that case, a post hoc test is conducted to compare it with a control method. This is done because the null hypothesis assumes no differences among the models. A paired comparison is carried out through a post hoc test to compare the differences among individual models. Our empirical experiment uses the Nemenyi test to demonstrate the difference when the average ranks differ by at least a Critical Difference (CD). The CD is calculated as follows:

$$CD = q_{a,\infty,k} \sqrt{\frac{k(k+1)}{12D}} \tag{8}$$

where

$$q_{a,\infty,k} \tag{9}$$

The Nemenyi test diagram shows the average ranks of the ADHE model at various levels of significance. The lines connecting different models indicate the average ranks of the ADHE model, and the number of datasets represented by D is used to calculate the Critical Difference (CD). The t-test statistic is used in this calculation.

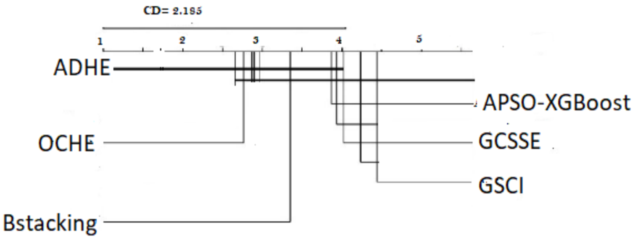

Figure 1. The CD diagram with the results of the Nemenyi test

Figure 1 presents the CD diagram that displays the results of the Nemenyi test. The diagram's horizontal axis represents the average rankings of the benchmark models for each dataset. A black box connects the benchmarks with a difference in average ranks lower than the CD value. The proposed ADHE is significantly better than OCHE, Bstacking, APSO-XGBoost, GCSSE and GSCI.

4. CONCLUSION

Machine learning techniques have shown great potential in accurately assessing creditworthiness, making them increasingly common in credit scoring models. In this study, new predictive models were developed to enhance machine learning's accuracy and ability to differentiate between creditworthy and non-creditworthy customers, facilitating faster credit decisions by financial institutions. Consequently, the financial industry has embraced machine learning algorithms to improve the accuracy of customer categorisation. The proposed adaptive dynamic heterogeneous ensemble model provides a faster and more efficient method for predicting customer credit scores and mitigating financial losses. In addition, the ensemble model includes supplementary metrics to support unbiased decision-making, addressing previous studies that emphasised the importance of multiple metrics in evaluating model performance.

However, the study faced certain limitations, such as the time-consuming nature of processing large datasets with grid search cross-validation, which necessitated the use of randomised cross-validation. The experiments were also conducted on Google Colab with less than six months of machine learning programming experience. This could have affected the results, which could be further improved by employing more efficient feature selection techniques besides PCA. The study could be replicated in the future with more imbalanced datasets and more powerful computers to determine whether better results can be obtained.

REFERENCES

[1] W. Frame, A. Srinivasan and L. Woosley, “The effect of credit scoring on small-business lending,” Journal of Money, Credit and Banking, vol. 33, no. 3, pp. 813-825, 2001.

[2] T. Tang, “Information asymmetry and firms' credit market access: Evidence from Moody's credit rating format refinement,” Journal of Financial Economics, vol. 93, no. 2, pp. 325-351, 2009.

[3] J. Crook, D. Edelman and L. Thomas, “Recent developments in consumer credit risk assessment,” European Journal of Operational Research, vol. 183, no. 3, pp. 1447-1465, 2007.

[4] A. Blöchlinger and M. Leippold, "Economic benefit of powerful credit scoring," Journal of Banking and Finance, vol. 30, no. 3, pp. 851-873, 2006.

[5] N. Chen, B. Ribeiro and A. Chen, "Financial credit risk assessment: a recent review," Artificial Intelligence Review, vol. 45, no. 1, pp. 1-23, 2016.

[6] A. El-Qadi, M. Trocan, T. Frossard and N. Díaz-Rodríguez, "Credit Risk Scoring Forecasting Using a Time Series Approach," in MaxEnt 2022, Basel Switzerland.

[7] A. El Qadi, M. Trocan, N. Díaz-Rodríguez and T. Frossard, "Feature contribution alignment with expert knowledge for artificial intelligence credit scoring," Signal, Image and Video Processing, vol. 17, no. 2, pp. 427-434, 2023.

[8] A. Aida, S. M. Shamsuddin and A. L. Ralescu, "Classification with class imbalance problem: a review," International Journal of Advances in Soft Computing and its Applications, vol. 5, no. 3, 2015.

[9] R. Adhao and V. Pachghare, "Feature selection using principal component analysis and genetic algorithm," Journal of Discrete Mathematical Sciences and Cryptography, vol. 23, no. 2, pp. 595-602, 2020.

[10] A. Asuncion and D. Newman, "UCI Machine Learning Repository," 2007.

[11] Z. Runchi, X. Liguo and W. Qin, "An ensemble credit scoring model based on logistic regression with heterogeneous balancing and weighting effects," Expert Systems with Applications, vol. 212, 2023.

[12] J. Mushava and M. Murray, "A novel XGBoost extension for credit scoring class-imbalanced data combining a generalized extreme value link and a modified focal loss function," Expert Systems with Applications, vol. 202, 2022.

[13] J. Mushava and M. Murray, "An experimental comparison of classification techniques in debt recoveries scoring: Evidence from South Africa's unsecured lending market," Expert Systems with Applications, vol. 111, pp. 35-50, 2018.

[14] Y. Wu, W. Huang, Y. Tian, Q. Zhu and L. Yu, "An uncertainty-oriented cost-sensitive credit scoring framework with multi-objective feature selection," Electronic Commerce Research and Applications, vol. 53, 2022.

[15] H. He, W. Zhang and S. Zhang, "A novel ensemble method for credit scoring: Adaption of different imbalance ratios," Expert Systems with Applications, vol. 98, pp. 105-117, 2018.

[16] W. Liu, H. Fan and M. Xia, "Credit scoring based on tree-enhanced gradient boosting decision trees," Expert Systems with Applications, vol. 189, 2022.

[17] Y. Xia, C. Liu, B. Da and F. Xie, "A novel heterogeneous ensemble credit scoring model based on bstacking approach," Expert Systems with Applications, vol. 93, pp. 182-199, 2018.

[18] R. M. Cruz, R. Sabourin and G. D. Cavalcanti, "META-DES.Oracle: Meta-learning and feature selection for dynamic ensemble selection," Information Fusion, vol. 38, pp. 84-103, 2017.

[19] L. Yang, "Classifiers selection for ensemble learning based on accuracy and diversity," in Procedia Engineering, 2011.

[20] G. U. Yule, "On the Association of Attributes in Statistics: With Illustrations from the Material of the Childhood Society, &c," Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character, pp. 257-319, 1900.

[21] L. L. Minku and X. Yao, "DDD: A New Ensemble Approach for Dealing with Concept Drift," IEEE Transactions on Knowledge and Data Engineering, vol. 24(4), pp. 619-633, 2012.

[22] T. Chen and C. Guestrin, "XGBoost," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 2016.

[23] J. Kennedy and R. Eberhart, "Particle swarm optimization," Proceedings of ICNN'95 - International Conference on Neural Networks, pp. 1942-1948, 1995.

[24] F. van den Bergh and A. Engelbrecht, "A new locally convergent particle swarm optimiser," IEEE International Conference on Systems, Man and Cybernetics, vol. 6, 2002.

[25] N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321-357, 2002.

[26] M. Mercier, M. S. Santos, P. H. Abreu, C. Soares, J. P. Soares and J. Santos, "Analysing the Footprint of Classifiers in Overlapped and Imbalanced Contexts," pp. 200-212, 2018.

[27] R. Wang and G. Liu, "Ensemble Method for Credit Card Fraud Detection," International Conference on Intelligent Autonomous Systems (ICoIAS), pp. 246-252, 2021.

[28] Y. Xia, C. Liu, Y. Li and N. Liu, "A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring," Expert Systems with Applications, vol. 78, pp. 225-241, 2017.

[29] C. Sano, "Japanese Credit Screening Data Set".

[30] "Statlog (German Credit Data) Dataset," UCI: Machine Learning Repository, 2023.

[31] Y. Xia, L. He, Y. Li, N. Liu and Y. Ding, "Predicting loan default in peer-to-peer lending using narrative data," Journal of Forecasting, vol. 39(2), pp. 250-280, 2020.

[32] J. Xiao, X. Zhou, Y. Zhong, L. Xie, X. Gu and D. Liu, "Cost-sensitive semi-supervised selective ensemble model for customer credit scoring," Knowledge-Based Systems, vol. 189, 2020.

[33] X. Chen, S. Li, X. Xu, F. Meng and W. Cao, "A Novel GSCI-Based Ensemble Approach for Credit Scoring," IEEE Access, vol. 8, 2020.

[34] C. Qin, Y. Zhang, F. Bao, C. Zhang, P. Liu and P. Liu, "XGBoost Optimized by Adaptive Particle Swarm Optimization for Credit Scoring," Mathematical Problems in Engineering, vol. 2021, pp. 1-18, 2021.

[35] S. Lessmann, B. Baesens, H.-V. Seow and L. C. Thomas, "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research," European Journal of Operational Research, pp. 124-136, 2015.

[36] J. Demšar, "Statistical Comparisons of Classifiers over Multiple Data Sets," Journal of Machine Learning Research, vol. 7, pp. 1-30, 1 December 2006.