ISSN 2063-5346

# Heterogeneous Ensemble Credit Scoring Model Using Multilevel Stacking

**Anil Kumar C J[1], B.K. Raghavendra[2]**

[1] *Associate Professor, Department of Computer Science and Engineering, ATME College of Engineering, Mysore, Karnataka, India.* anilkumarcj@gmail.com

[2] *Professor & Head, Department of Information Science and Engineering, Don Bosco Institute of Technology, Bangalore, Karnataka, India.* bkraghavendra@dbit.co.in

Author email: anilkumarcj@gmail.com

**Article History: Received:** 12.05.2023 **Revised:** 25.05.2023 **Accepted:** 05.06.2023

## Abstract

Creating sophisticated credit scoring models is a useful method of locating defaulters. Extensive study was done on the ensemble credit rating since the ensemble learning approaches proven to be more effective than the individual classifiers. There are no standardised benchmark features for bank clients' credit scores, and each country and bank decides which standards to use based on the information it has available about its customers. Therefore, using the same classifiers on several data sets may yield excellent results for certain data sets but not others. Combining the best classifiers from among those available for each dataset independently with the ability to diversify the classifiers in ensemble learning methods produces excellent results for all datasets. In this work, we developed 6 different multilevel stacking method in which we carried out the credit scoring using 3 base classifiers at level 0, 3 meta-classifiers at level 1 and MLP as final classifier at level 2. The German and Australian UCI data sets were used to evaluate the suggested methodology. The ensemble models that combined KNN+DF+NB in level-1 and SVM+LR+RF in level-2 and MLP as final meta-learner at level 3 exhibited the best accuracy, AUC, and F1-score performance. The findings of this study showed that both balanced and unbalanced data sets produce highly favourable results using the suggested methodology.

**Keywords:** Ensemble model, KNN, DT, NB, LR, RF, ANN, SVM, Credit scoring

## 1. Introduction

Basel II states that a bank's capacity to store capital is influenced by its credit default risk and connected with its rating. One of the two basic techniques for determining and monitoring credit risk under the agreement is the use of credit rating agencies. [1]. A credit scoring model's goal is to predict a loan applicant's likelihood of default based on historical data [2]. It helps lenders decide whether to approve or reject borrowers. Numerous research on customer credit scoring have been conducted as a result of the significance of credit risk assessment for banks. A scoring model was utilised in order to arrive at estimates for the probability of default (PD), exposure at default (EAD), and loss-given default (LGD). Among these methods, the application of PD models has grown in popularity as a study subject.

There are two main categories of scoring models: those that use artificial intelligence and those that rely on statistical techniques. Logistic regression [3], linear discriminate analysis (LDA), and k-nearest neighbour are statistical methods (KNN). Artificial intelligence methods include decision trees, support vector machines, boosting algorithms, and artificial and deep neural networks (ANN & DANN) [4]. Numerous scholars have also suggested the best credit scoring techniques. A more advanced multi-population niche genetic algorithm for credit scoring was proposed by [5] and [6] as an information gain driven genetic algorithm wrapper feature selection for credit rating. [7] suggested an ensemble strategy for credit scoring based on multi-level classification and modified PSO clustering. An expert credit rating model was optimised by [8] using a genetic algorithm. Another technique in the literature for determining credit scores is the robust optimization method [9]. The usage of single classifiers and ensemble classifiers, which combine two or more classifiers, can be used to categorise credit scoring methodologies. One machine learning algorithm might not produce the greatest outcomes. For credit scoring issues, recent studies established ensemble models, which produce more precise and sophisticated models than single classification techniques. As demonstrated by [10], the final classification process can be enhanced by integrating or fusing the data collected from various classifiers. An increase in the classification task's accuracy and resilience might be used to illustrate this. As long as the variety of the basic classifiers does not reduce the accuracy of the ensemble members, an ensemble system can be improved [11].

Voting, bagging, boosting, and stacking are now the most popular ensemble procedures for credit scoring. Due to the stacking approach's great performance and resilience, it has been widely utilised in credit scoring models [12]. Although the stacking method used to build ensemble models demonstrated superiority, it is crucial to consider the performance of the base classifiers in order to maximise the stacking's benefits [5]. [13] suggested a hybrid ensemble credit scoring model that makes use of four classification algorithms and feature selection approaches. The classification techniques used are DT, SVM, NB, and generalised linear models (GLM). 8 ensemble models were created as a consequence, including GLM+SVM+NB+DT, GLM+SVM+NB, GLM+SVM+DT, GLM+NB+DT, SVM+NB+DT, GLM+DT, and SVM+DT. Among these 8 models, GLM+DT worked well with higher accuracy. By utilising a hybrid ensemble credit scoring model with stacking-based noise detection and weight assignment, one can lessen the detrimental effects of noisy data on classification methods [12]. The noise-detected training data should be utilised to generate noise-detected training data and remove or adapt noisy data from raw datasets in order to increase default risk prediction competence. [14] has suggested a hybrid ensemble credit scoring model to address the issue of unbalanced data classification. This proposed model, which is built on the LSVM, KNN, MDA, DT, and LR classifiers, adaptively chooses the one with the highest AUC in accordance with the data distribution, then integrates all of the underlying classifiers to produce a forecast. [15] built a heterogeneous ensemble classifier using base classifiers including Naive Base, Support Vector Machine, Decision Tree, Random Forest, and Logistic Regression. These six base classifiers are used to create the proposed model's ensemble aggregation. Specifically, the random forest technique is used to choose the best features. The ensemble model is built using the stacking and voting technique.

The effectiveness of ensemble models over single classifiers has already been mentioned. However, by boosting variety, stacking-based ensemble models' outputs can be enhanced. The challenge of choosing the best algorithm is presented by the inclusion of various algorithms. Voting techniques are frequently used in ensemble learning issues to combine algorithms. Combining the algorithms and weighting them in accordance with an optimization procedure can lead to better outcomes. However, some unnecessary or redundant features may be present in credit scoring datasets, lengthening training time and lowering algorithm performance. The complexity of the algorithms can be kept to a minimum, training time can be cut down, and accuracy can be raised by the use of FS [16].

This study used a variety of base classifiers, including Naive Bayes (NB), Support Vector Machines (SVM), Decision Trees (DT), K-Nearest Neighbors (KNN), Logistic Regression (Logit), and Random Forest (RF). The proposed methodology was evaluated using three performance metrics, including Accuracy, F-Measure and AUC, and two benchmark datasets of UCI Australian and Germany.

## 2. Classification Techniques

### 2.1 Artificial neural network (ANN)

Artificial neural network (ANN) models frequently use a non-parametric method that was influenced by the structure of a neuron in order to capture complex connections between inputs and outputs [17]. The type of ANN that is most frequently used is the multi-layer perceptron (MLP), which has one input layer, one or more hidden layers, and one output layer. ANN has been used to handle classification problems and credit risk prediction (regressions) concerns [18] [19]. The ANN model begins passing the features of each consumer to the input layer. Prior to arriving to the output layer, which displays the final prediction based on the weights, the hidden layers process these features. The latter is defined for each feature according to its relative relevance. Finally, using an activation function like the sigmoid, all the weighted features are merged to form outputs. This strategy is applied repeatedly to reduce the difference between the anticipated and true class [20].

The framework of an ANN is designed after that of the biological brain system. Credit scoring is just one of many areas where ANN can be put to good use; Other fields include signal processing, time series prediction, pattern recognition, data categorization and clustering, etc. Modifying the connections between the neurons is part of training an ANN. The system's topology determines whether an ANN is a feedforward neural network (FFNN) or a feedback neural network (FBNN). The data stream in an FFNN network is unidirectional and does not contain any response loops, but in an FBNN network, the stream is bidirectional and does contain response loops. Multilayer Perceptron (MLP), a three-layer network design with an input layer, a hidden layer, and an output layer, is a member of the FFNN family. The layer that receives the inputs has the same number of neurons as features in the dataset. The hidden layer of the MLP is essential because it serves as a mapping between the input and output layers. The output layer is used to give the network's final result.

Each neuron in an MLP network is equipped with summation and activation functions, and all neurons in the layer are connected to one another by numerical weights. The inputs, weights, and bias are disclosed in Eq, and the summation function provides a concise summary of the Equation (1). Wij is the weight of the connection between input neuron i and output neuron j, bj is a bias term, and n is the total number of neuron inputs. The output of the summing function will be sent into the activation functions. S-shaped sigmoid functions are frequently utilised for non-linear activation functions. Eq. displays the sigmoid Equation (2). Therefore, the output of neuron j can be represented as in Equation (3).

$$S_j = \sum_{i=1}^{n} w_{ij}\, I_i + \beta_j \qquad (1)$$

$$f(x) = \frac{1}{1+e^{-x}} \qquad (2)$$

$$y_j = f_j\left(\sum_{i=1}^{n} w_{ij} I_i + \beta_j\right) \qquad (3)$$

When an ANN is being built, its parameters can be optimised with the help of this method's learning procedure (set of weights). Instead, the outcomes are estimated using a curved and updated weighting system, which also helps to lower the standard deviation of the estimated errors. Supervised learning is a key part of MLP training. Supervised learning seeks to reduce the discrepancy between predicted and actual results. Backpropagation is one of the most widely used supervised learning techniques based on Gradient. Learning the derivative of an ANN's objective function in terms of the weights and biases that replace between layers is an iterative process. The method struggles to perform well when the search space is huge and excels only with discriminable goal functions.

## 2.2 K-Nearest Neighbour

The K-nearest-neighbor (KNN) model is a type of statistical learning that does not rely on parameters. Classification, prediction [21], audio-visual recognition, and many other cutting-edge applications all make heavy use of it in AI research. The key function of the KNN model is to assign a category to an unlabeled data point by comparing it to those of known data points. It begins by extracting the characteristics of the data that will be classified and comparing them to the testing set's data from recognised categories. Then, it chooses the components that are the closest to the real point, and it calculates the frequency of the category. The closest and most comparable neighbour is used as the final step [22]. The Manhattan distance, the Euclidean distance, and the Chebyshev distance are just a few of the distance calculating methods available. Because of its greater accuracy for high-dimensional data, the Manhattan distance is used in this research. Two points, xi and xj, each having characteristics I and j, are separated by a Manhattan distance, denoted by dM [23], is given in Equation (4):

$$d_M\,(xi, xj) = \sum |xi - xj| \qquad (4)$$

## 2.3 Support Vector Machine

Classification and regression problems are addressed by employing a sophisticated supervised learning model called the support vector machine (SVM) [24]. SVM is different from neural networks in that it is based on structural risk minimization (SRM) ideas, which promote better generalisation (NN). Using support vectors, SVM recognises patterns between two classes of points (SV). The decision surface known as SV is obtained by solving a quadratic programming problem [25]. As stated in Equation (5), the primary SVM task is to estimate a classification function.

$$f: R^n \rightarrow \{\pm 1\} \qquad (5)$$

where f is the function that maps points x to their correct classification y; the input/output training data are from classes (x1, y1), . . . , (xi, yi) ∈ $R^n$ × {±1}. Equation (6) provides the SVM equation:

$$f(x) = \sum_i^{n=1} y_i \alpha_i k(x, x_i) + b \qquad (6)$$

As shown in Equation (7), the kernel function transfers the lower-dimensional space and returns the dot product of the transformed vectors in the higher-dimensional space (transformed space), where (xi, yi) is the ith training point, αi and b are the learning weights, and k(x, xi) is the kernel function.

$$k(x, x_i) = \phi(x)\cdot\phi(x_i) \qquad (7)$$

### *2.4. Logistic Regression*

Logistic regression (LR) is the statistical model that employs a logistic function in its most fundamental form to represent a binary dependent variable, despite the existence of many more complex extensions. A binary logistic model uses an indicator variable with the possible values "0" and "1" to represent a dependent variable with two possible mathematical values, such as pass or fail. The decision function is constrained by a machine learning model to a particular set of circumstances. The model's hypothesis space is determined by this set of requirements. Of course, we also expect that these requirements are straightforward and sensible. The LR model's presumptions are as follows:

$$P(y = 1|x; \theta) = g(\theta^T x) = \frac{1}{1+ e^{-\theta^T * x}} \quad (8)$$

The decision function that corresponds to the sigmoid function, denoted by g(h), is:

$$y^* = 1, if\ P(y = 1|x) > 0.5$$

The standard approach is to set the threshold at 0.5. You can choose from a variety of thresholds in the real application. You can pick a higher threshold if the positive example's accuracy is high. You can choose a lower threshold if recall is high.

### *2.5. Random Forest*

An ensemble learning-based categorization method is Random Forest. The DT classifier is used to construct each classifier in the ensemble. It is a group of classifiers that come together to form a forest. The attributes used to build each decision tree are chosen at random for each node. Every tree casts a vote during classification. Based on votes, the class with the most votes is taken into account in the output.

Training a random forest typically involves the use of bagging or Bootstrap aggregation methods. Consider a training set xi ∈X with class output yi∈Y. Bagging employs a series of random sample selections to ensure that the decision trees are a good match for the training dataset. Attribute-based tree-building results in several trees, each of which uses a unique classification for newly introduced instances. An algorithm uses votes to decide which category a new instance best fits into.

The RF classifier uses the Gini index to decide which features to use. The Gini index evaluates how impure an attribute is in comparison to a certain class. When one case is chosen at random from a training set, the Gini index is;

$$\sum\sum(f(C_{i,}T)/|T|)\ (f\ (C_j, T)/|T) \qquad (9)$$

The probability of the chosen case Ci is represented by the expression f (Ci, T) / |T|.

### 2.6. Decision Tree

The decision tree is a common supervised machine learning technique for solving classification and regression problems (DT). It is renowned for having a structure like a tree that is simple to understand when visualised as a tree. Leaf nodes and internal nodes make up DT. The leaf nodes represent the final class, whereas the inner nodes represent tests over attributes and have multiple branching for each possible conclusion. The best qualities are determined using a heuristic or statistical method as part of the top-down divide-and-conquer construction of DT. The information gain is calculated after each cycle to decide whether or not to add a feature. The mathematical formula for determining the information gain is shown in equation (10) where $\sum_{i=1}^{c} p_i log_2(p_i)$ calculates the entropy, where Pi stands for the instance (i) with the highest probability and c stands for the classes that are present in the dataset. Additionally, S stands for the sample set, Sv for the elements that hold v in the attribute A, V(A) for the values of the variable A, and E for entropy.

$$IG(S,A) = -\sum_{i=1}^{c} p_i log_2(p_i) - \sum_{v \in V(A)}^{c} \frac{S_v}{S} E(S_v) \quad (10)$$

### 2.7. Naïve Bayes

The Naive Bayesian (NB) algorithm relies on the application of the Bayesian theorem with high independence requirements among the characteristics. The necessary probability terms for classification are estimated using a set of training data. The precision of the predictable needed probability terms serves as a gauge for this performance.

Conditional probability is the major emphasis of the naive Bayes classifier. It is appropriate for high dimensional inputs and makes the assumption that the characteristics and features are independent. The presumption is that, given the instance's target value, the product of the probabilities for each individual attribute determines the likelihood that the conjunction $a_1$, $a_2$, $a_n$ will be noticed.

$$P\left(a_1, a_2, \dots, a_n \middle| v_j\right) = \prod_i P(a_i | \, v_j) \quad (11)$$

In this case, ai stands for a distinct attribute value, and vj for a distinct target value.

### 2.8 Stacking Learning

It is crucial to mix various machine learning models to create a learning model since some machine learning models might not perform well when faced with the demands of challenging jobs. The ensemble learning model is the name of this teaching strategy. Two categories can be used to categorise ensemble learning. The first is referred to as the sequential method, in which machine learning models, including boosting and gradient boosting, are built consecutively and have a strong dependency on one another. The second approach is referred to as the parallel technique, in which machine learning models like bagging and random forest are created in parallel without a heavy reliance [26]. A distinct ensemble learning model from boosting and bagging is stacking [27]. Stacking enhances the model's functionality, lowers the generalisation error, and makes the model more widely applicable. Through the use of a meta-classifier, stacking integrates many machine learning models. There are two main layers in the stacking model. Base classifiers, a kind of machine learning models, make up the top layer. With varying classification performances, the base classifiers go on with the output prediction process using the input data. All of the output predictions are combined during stacking to create fresh inputs for the second layer. The final prediction is obtained by the second layer, which is known as the meta classifier, using the new inputs as its input.

### 2.8.1 Multi-level Stacking

Multi-level stacking is refined by applying it to several stacked layers.

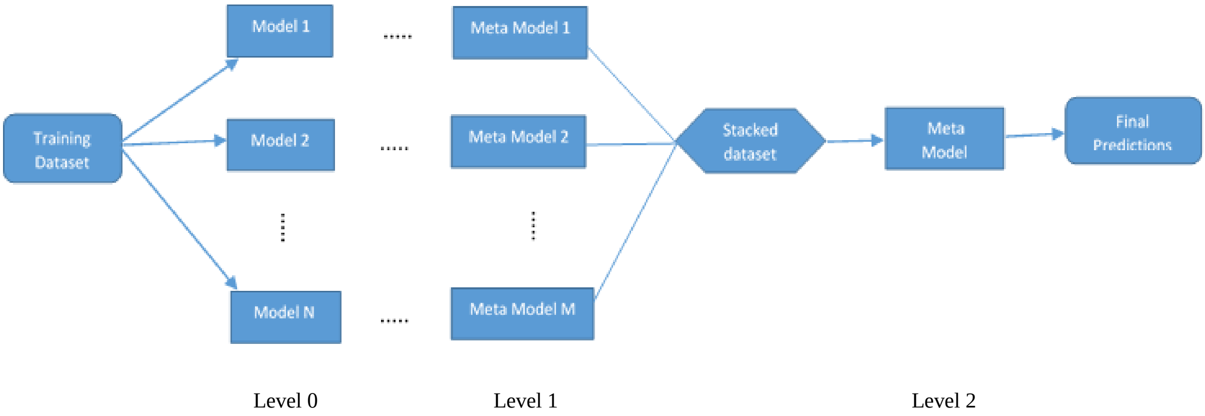

Figure 1: Multilevel Stacking

In a three-level stacking, for instance, Level-0 is the same since different base learners are taught using k-fold cross validation. N of these meta models are used at level 1 rather than just one. The N meta-models of level-1's final meta model are used at level-2 to make predictions.

## 3 Experimental study

### *3.1. Credit Datasets*

The experiments make use of two real-world datasets from the UCI machine learning library, namely the German and Australian datasets. In Table 1, the datasets' specifics are displayed.

There are 1000 samples total in the German credit dataset, 700 of which are positive and 300 of which are negative. Each instance has a target attribute, 13 categories features, and 7 numerical features. Among the 690 samples in the Australian dataset, 307 samples are positive and 383 samples are negative. Each instance consists of a target attribute, six categorical attributes, and eight numerical features.

**Table1.** Credit datasets taken from the UCI machine learning repository

| **Dataset** | **Number of Instances** | **Good cases** | **Bad cases** | **Categorical features** | **Numeric features** | **Total features** |
|---|---|---|---|---|---|---|
| German | 1000 | 700 | 300 | 13 | 7 | 20 |
| Australian | 690 | 307 | 383 | 6 | 8 | 14 |

### *3.2. Feature Selection*

Improving credit scoring system predictions with feature selection's inexpensive and quick classifiers is possible. It is possible to combine the feature selection and subset selection processes. Wrappers, filters, and embedding methods are the three primary groups from which to choose which features to use.

The random forest approach is a method that is based on trees, and we have utilized it in this study to choose features. The random forest approach is used to determine the value of each characteristic, after which the least relevant features are eliminated and the most significant elements are chosen. As a method of classification, random forest is frequently utilized. It also has the ability to determine the relevance of the features, and as a result, it can function as a selector for those features.

The functioning of random forest is based on constructing a series of decision trees. The final significant feature Aj can be calculated as the following given a difference in the performance for the tree i denoted by $d_i$.

$$I(A_j) = \sum d_i/(nxSE_d) \qquad (12)$$

Where n is the number of elements in the dataset, $SE_d$ is the standard error of $d_i$ taking into account all trees ($SE_d$ = SDdi$\sqrt{n}$ ), and $SD_d$ is the standard deviation of $d_i$.

### *3.3. Proposed Method*

For the purpose of ensemble aggregation, the suggested model is constructed with the help of six different base classifiers. In order to choose the most useful characteristics, a feature selection algorithm that makes use of the random forest method is utilized. The dataset pertaining to credit is split up into a training set and a test set. The multilevel stacking method is utilized for ensemble classification in the model that was suggested. Stacking is first used on the base classifiers of the classification system. This is done in two levels. To begin, the training dataset is subdivided into 10 folds so that it can be subjected to cross validation. Each iteration used 10-1 (9) folds to train the baseline classifiers and 1 fold to predict the result. The complete training set's prediction is obtained after 10 iterations. Meta-features, or the predictions from each classifier, are collected from the output of the classifiers and added to the dataset. The second stage makes use of three unique MCs, or meta-classifiers. The output of these meta-classifiers is predicted using MLP as a final mete-learner.

### 3.4. Evaluation Measures

The choice of evaluation metrics is crucial for confirming the effectiveness of the categorization models. For various assessment and prediction metrics, confusion matrices have been taken into consideration. Table 2 displays the confusion matrix.

**Table 2. Confusion Matrix**

| | | **Predicted** | |
|---|---|---|---|
| | | **Positive** | **Negative** |
| **Real** | **Positive** | True Positive (TP) | False Negative (FN) |
| | **Negative** | False Positive (FP) | True Negative (TN) |

Instances that fall inside the positive class and are accurately classified as positive by the model are known as true positives (TP). False positives (FP) are situations that the model mistakenly classifies as positive even when they actually belong in the negative category. False negatives (FN) are instances that belong to the positive class but are incorrectly identified as negative by the model. Finally, negative cases that were correctly labelled as negative by the model are called true negatives (TN). Accuracy, Precision, Recall, F-measure, Specificity, and Area Under the ROC Curve (AUC) are stated using the confusion matrix as follows:

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \tag{13}$$

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

$$Recall = Sensitivity = \frac{TP}{TP+FN} \tag{15}$$

$$F - Measure = \frac{2\ XPrecisionXRecall}{Precision+Recall} \tag{16}$$

The F-measure, as shown in equation (16), is a way to evaluate the accuracy of a model. It is calculated by taking twice the product of recall and precision, and then dividing by the sum of precision and recall. The number of true positive outcomes divided by the total of true positive and false positive results is the definition of precision in equation (14). Recall is calculated by dividing the total of true positive and false negative findings by the number of true positive outcomes. The Receiver Operating Characteristic (ROC) curve's two-dimensional area under the curve is measured by the AUC, or area under curve. AUC values above a certain threshold imply improved model performance.

### 4. Results and Discussion

Eighty percent of the dataset is used as training data, while twenty percent is used for testing. As base models, we utilise SVM, LR, KNN, RF, Naive Base, and DT, and MLP as the final meta-learner.

We evaluate a multi-layer stacking ensemble. The ensemble uses multiple meta classifiers on different layers. Our objective is to create a stacking model (multilayer). Six different stacking models are created using 3 base classifiers at level 0, 3 meta-classifiers at level1 and one final meta-classifier at level 2.

The testing set, which included datasets from both Australia and Germany, served as the foundation for the findings in this section. Each classifier is trained using ten-fold cross validation.

The results of several ensemble classifiers are displayed in Tables 3 and 4 with respect to Accuracy, F-measure, and AUC.

**Table 3. The classification results for the Australian dataset.**

| First Layer | Second Layer | Meta Classifier | Accuracy | F-Measure | AUC |
|---|---|---|---|---|---|
| KNN+LR+DT | SVM+RF+NB | MLP | 90.35 | 89.15 | 88.36 |
| **KNN+DT+NB** | **SVM+LR+RF** | **MLP** | **94..23** | **94.02** | **93.47** |
| DT+NB+SVM | KNN+LR+RF | MLP | 88.65 | 88.56 | 87.35 |
| RF+SVM+KNN | DT+LR+NB | MLP | 87.95 | 87.84 | 86.69 |
| LR+SVM+RF | KNN+DT+NB | MLP | 87.62 | 87.47 | 86.64 |
| LR+RF+DT | SVM+NB+KNN | MLP | 91.56 | 89.45 | 88.48 |

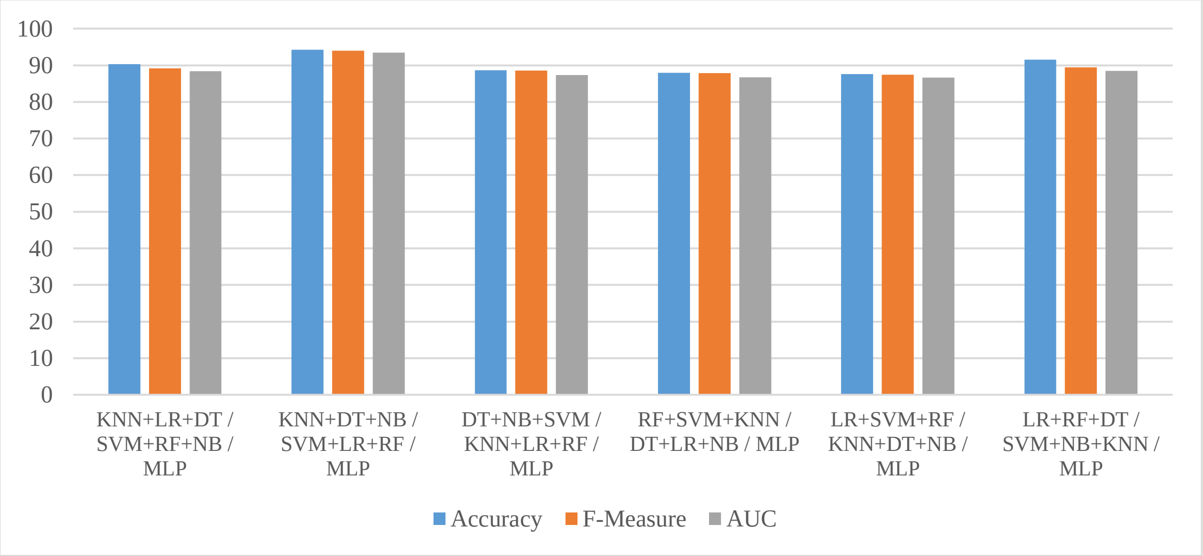

**Figure 2: Results of various multilayer stacking models using various classifier placements on an Australian dataset**

**Table 4. The classification results for the German dataset**

| First Layer | Second Layer | Meta Classifier | Accuracy | F-Measure | AUC |
|---|---|---|---|---|---|
| KNN+LR+DT | SVM+RF+NB | MLP | 84.35 | 84.15 | 83.36 |
| **KNN+DT+NB** | **SVM+LR+RF** | **MLP** | **87..23** | **87.02** | **86.47** |
| DT+NB+SVM | KNN+LR+RF | MLP | 86.65 | 85.56 | 85.35 |
| RF+SVM+KNN | DT+LR+NB | MLP | 84.95 | 83.84 | 82.69 |
| LR+SVM+RF | KNN+DT+NB | MLP | 83.62 | 83.47 | 82.64 |
| LR+RF+DT | SVM+NB+KNN | MLP | 82.95 | 82.15 | 81.69 |

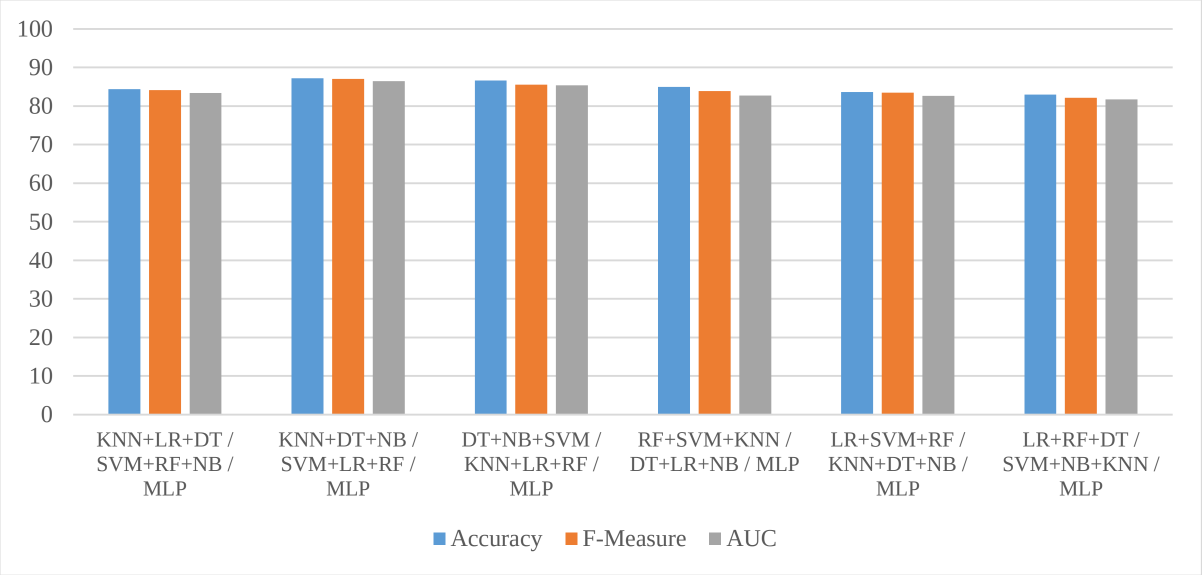

**Figure 3: Results of various multilayer stacking models using various classifier placements on a German dataset**

The ensemble models that incorporated KNN+DF+NB in level 1, SVM+LR+RF in level 2, and MLP as the final meta-learner at level 3 had the highest performance in terms of accuracy, AUC, and F-measure out of all the stacking models we developed.

**Table 5. Comparison of performance with other credit scoring models**

| Reference | Method | Australian dataset | German dataset |
|---|---|---|---|
| Tripathi (2018) [28] | NB-DT-QDA-TDNN-PN N (Layered weighted voting) | 93.02 | 84.26 |
| Xia et.al (2020) [29] | RF + GBDT + XGBoost + LightGBM + CatBoost | 86.39 | 77.72 |
| Yotsawat et.al (2021) [2] | Cost-sensitive neural network ensemble | 84.93 | 74.40 |
| Zou et.al (2022) [30] | Extreme learning machine enhanced gradient Boosting | 86.35 | 76.17 |
| Nasser Khalil (2022) [31] | Integrated optimal hybrid ensemble credit scoring model based on classifier Selection | 93,21 | 84.53 |
| Proposed method | KNN+DF+NB / SVM+LR+RF / MLP | **94..23** | **87.23** |

## 5. Conclusion

Effective analysis of default customers is essential to enhancing the financial standing of banks and other financial organisations. Creating sophisticated credit scoring models is a useful method of locating defaulters. In the credit risk literature, a variety of methods have been proposed for credit scoring of bank customers. Among these methods are single classifier methods and multiple classifier methods which combine to perform ensemble credit scoring. The effectiveness of ensemble learning approaches over single classifiers has led to substantial study on ensemble credit scoring and the development of a number of different techniques. On the other hand, there are no benchmark features that can be utilised globally for the credit score of bank clients, and each nation and bank chooses which aspects to use based on the facts about its consumers. As a result, using the same classifiers on various datasets may yield excellent results for some but not others. Combining the best classifiers from among those available for each dataset independently with the ability to diversify the classifiers in ensemble learning methods produces excellent results for all datasets.

In this work, we developed 6 different multilevel stacking method in which we carried out the credit scoring using 3 base classifiers at level 0, 3 meta-classifiers at level 1 and MLP as final classifier at level 2. the ensemble models that combined KNN+DT+NB in level-1 and SVM+LR+RF in level-2 and MLP as final meta-learner at level 3 exhibited the best accuracy, AUC, and F1-score performance. Additionally, the random forest feature selection approach is used to carry out the feature selection process.

The UCI German and Australian datasets were used to evaluate the suggested technique. The study's findings demonstrated that the suggested approach produces very favourable outcomes for both balanced datasets (with roughly equal numbers of excellent and bad classes) and unbalanced datasets (with a large difference between the number of good and bad classes). This method can therefore be used to the internal credit scoring systems of some developing nations without credit scoring agencies as well as the credit scoring of real data by credit scoring companies.

Future research directions focus more on using different classifiers as meta-classifiers at level 2 in stacking and also extending this to different datasets.

## REFERENCES

1. Doumpos, M., Lemonakis, C., Niklis, D., & Zopounidis, C. (2019). Analytical techniques in the assessment of credit risk. New York: Springer.
2. Yotsawat, W., Wattuya, P., & Srivihok, A. (2021). A Novel Method for Credit Scoring based on Cost-sensitive Neural Network Ensemble. IEEE Access.
3. Dumitrescu, E., Hue, 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.
4. Zhang, A., Peng, B., Chen, J., Liu, Q., Jiang, S., & Zhou, Y. (2022). A ResNet-LSTM Based Credit Scoring Approach for Imbalanced Data. Mobile Information Systems, 2022.
5. Zhang, W., He, H., & Zhang, S. (2019). A novel multi-stage hybrid model with enhanced multi-population niche genetic algorithm:

An application in credit scoring. Expert Systems with Applications, 121, 221-232.

6. Jadhav, S., He, H., & Jenkins, K. (2018). Information gain directed genetic algorithm wrapper feature selection for credit rating. Applied Soft Computing, 69, 541-553.
7. Singh, I., Kumar, N., Srinivasa, K. G., Maini, S., Ahuja, U., & Jain, S. (2021). A multi-level classification and modified PSO clustering based ensemble approach for credit scoring. Applied Soft Computing, 111, 107687.
8. Estran, R., Souchaud, A., & Abitbol, D. (2022). Using a genetic algorithm to optimize an expert credit rating model. Expert Systems with Applications, 203, 117506.
9. López, J., & Maldonado, S. (2019). Profit based credit scoring based on robust optimization and feature selection, Information sciences, 500, 190-202.
10. Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine learning, 40(2), 139-157.
11. Abellán, J., & Castellano, J. G. (2017). A comparative study on base classifiers in ensemble methods for credit scoring. Expert systems with applications, 73, 1-10
12. Jin, Y., Zhang, W., Wu, X., Liu, Y., & Hu, Z. (2021). A Novel Multi-Stage Ensemble Model With a Hybrid Genetic Algorithm for Credit Scoring on Imbalanced Data. IEEE Access, 9, 143593-143607.
13. Nalić, J., Martinović, G., & Žagar, D. (2020). New hybrid data mining model for credit scoring based on feature selection algorithm and ensemble classifiers. Advanced Engineering Informatics, 45, 101130.
14. Zhang, T., & Chi, G. (2021). A heterogeneous ensemble credit scoring model based on adaptive classifier selection: An application on imbalanced data. International Journal of Finance & Economics, 26(3), 4372-4385.
15. Anil Kumar, C. J., Raghavendra, B. K., & Raghavendra, S. (2022). A Credit Scoring Heterogeneous Ensemble Model Using Stacking and Voting. Indian Journal of Science and Technology, 15(7), 300-308.
16. Behr, A., & Weinblat, J. (2017). Default patterns in seven EU countries: A random forest approach. International Journal of the Economics of Business, 24(2), 181-222.
17. Bishop, C. M. (1995). Neural networks for pattern recognition. New York, NY: Oxford University Press Inc.
18. West, D. (2000). Neural network credit scoring models. Computers & Operations Research, 27(11), 1131–1152.
19. Mai, F., Tian, S., Lee, C., & Ma, L. (2019). Deep learning models for bankruptcy prediction using textual disclosures. European Journal of Operational Research, 274(2), 743–758.
20. 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.
21. Jamjoom, M.; Alabdulkreem, E.; Hadjouni, M.; Karim, F.; Qarh, M. Early Prediction for At-Risk Students in an Introductory Programming Course Based on Student Self-Efficacy. Informatica 2021, 45, 6.
22. Zhang, C.; Zhong, P.; Liu, M.; Song, Q.; Liang, Z.; Wang, X. Hybrid Metric K-Nearest Neighbor

algorithm and Applications. Math. Probl. Eng. 2022, 2022, 8212546.

23. Szabo, F. The Linear Algebra Survival Guide: Illustrated with Mathematica; Academic Press: cambridge, MA, USA, 2015.
24. Drucker, H.; Burges, C.J.; Kaufman, L.; Smola, A.; Vapnik, V. Support vector regression machines. In Advances in Neural Information Processing Systems; MIT Press: Cambridge, MA, USA, 1996.
25. Yue, S.; Li, P.; Hao, P. SVM classification: Its contents and challenges. Appl. Math. A J. Chin. Univ. 2003, 18, 332–342.
26. Tang, Y.; Gu, L.;Wang, L. Deep Stacking Network for Intrusion Detection. Sensors 2022, 22, 25.
27. Wolpert, D.H. Stacked generalization. Neural Netw. 1992, 5, 241–259
28. Tripathi, D., Edla, D. R., & Cheruku, R. (2018). Hybrid credit scoring model using neighborhood rough set and multilayer ensemble classification. Journal of Intelligent & Fuzzy Systems, 34(3), 1543-1549
29. Xia Y, Zhao J, He L, Li Y, Niu M. A novel tree-based dynamic heterogeneous ensemble method for credit scoring. Expert Systems with Applications. 2020;159:113615–113615. Available from: https://dx.doi.org/10.1016/j.eswa.2020.113615.
30. Zou, Y., & Gao, C. (2022). Extreme Learning Machine Enhanced Gradient Boosting for Credit Scoring. Algorithms, 15(5), 149.
31. Khalili, Nasser and Saleh Sedghpour, Alireza, An Integrated Optimal Hybrid Ensemble Credit Scoring Model Based on Classifier Selection. Available at http://dx.doi.org/10.2139/ssrn.4247190