Credit Risk Scoring: A Stacking Generalization Approach

Bernardo Raimundo[1], Jorge M. Bravo[1,2,3,4*][0000-0002-7389-5103]

[1] NOVA Information Management School (NOVA IMS), Campus de Campolide, Universidade Nova de Lisboa, 1070-312 Lisboa, Portugal

[2] Université Paris-Dauphine PSL, Place du Maréchal de Lattre de Tassigny

75775 PARIS Cedex 16

[3] ISCTE-IUL, Avenida das Forças Armadas, 1649-026 Lisboa

[4] Universidade de Évora, Rua do Cardeal Rei, 7000-849 Évora

Largo dos Colegiais, Nº 2, 7004-516 Évora

[*]Corresponding author

This is the Author Peer Reviewed version of the following conference paper published by Springer:

Raimundo, B., & Bravo, J. M. (2024). Credit Risk Scoring: A Stacking Generalization Approach. In Á. Rocha, H. Adeli, G. Dzemyda, F. Moreira, & V. Colla (Eds.), Information Systems and Technologies: WorldCIST 2023, Volume 1 (Vol. 1, pp. 382-396). (Lecture Notes in Networks and Systems; Vol. 799). Springer. https://doi.org/10.1007/978-3-031-45642-8_38

Credit Risk Scoring: A Stacking Generalization Approach

Bernardo Raimundo [1] and Jorge M. Bravo [2]

[1] NOVA IMS - Universidade Nova de Lisboa,
M20200240@novaims.unl.pt
[2] NOVA IMS - Universidade Nova de Lisboa & MagIC & Université Paris-Dauphine PSL & ISCTE-IUL Business Research Unit (BRU-IUL) & CEFAGE-UE, Lisbon, Portugal, ORCID: 0000-0002-7389-5103
jbravo@novaims.unl.pt

Abstract. Forecasting the creditworthiness of customers in new and existing loan contracts is a central issue of lenders' activity. Credit scoring involves the use of analytical methods to transform historical loan application and loan performance data into credit scores that signal creditworthiness, inform, and determine credit decisions, determine credit limits, and loan rates, and assist in fraud detection, delinquency intervention, or loss mitigation. The standard approach to credit scoring is to pursue a "winner-take-all" perspective by which, for each dataset, a single believed to be the "best" statistical learning or machine learning classifier is selected from a set of candidate approaches using some method or criteria often neglecting model uncertainty. This paper empirically investigates the predictive accuracy of single-based classifiers against the stacking generalization approach in credit risk modelling using real-world peer-to-peer lending data. The findings show that stacking ensembles consistently outperform most traditional individual credit scoring models in predicting the default probability. Moreover, the findings show that adopting a feature selection process and hyperparameter tuning contributes to improving the performance of individual credit risk models and the super-learner scoring algorithm, helping models to be simpler, more comprehensive, and with lower classification error rates. Improving credit scoring models to better identify loan delinquency can substantially contribute to reducing loan impairments and losses leading to an improvement in the financial performance of credit institutions.

Keywords: Credit scoring, Ensemble learning, Probability of default, Stacking generalization, Risk management.

1 Introduction

The critical role of the credit market in causing the most recent global financial crisis led to a strengthening in banking regulation and increased academic research and policy interest in this area. The banking and accounting regulatory changes brought by the revised Basel Committee on Banking Supervision Accords and the International Financial Reporting Standards (IFRS) #9 and Financial Accounting Standards Board (FASB) Current Expected Credit Loss (CECL) standards introduced stronger risk management

obligations for banks, with capital requirements closely linked with estimated credit portfolio losses over the entire life of the exposures, conditional on macroeconomic factors, on a point-in-time basis [1,2]. As a result, banks now devote substantial resources to developing internal credit risk models to better support decisions when granting loans, quantify expected credit losses, and assign mandatory economic capital [3,4]. Credit scoring involves the use of analytical methods to transform traditional historical and non-conventional loan application and loan performance data into credit scores that signal creditworthiness, inform, and determine credit decisions, determine credit limits, and loan rates, assist in fraud detection, delinquency intervention (e.g., loan restructuring) or loss mitigation. Originally credit scoring used a pure judgmental approach to accept or reject an application, known as the 5Cs of credit, composed of the following metrics: (i) the character of the applicant, (ii) the capital, (iii) the collateral, (iv) the capacity and (v) the condition [5]. However, due to its subjectivity, only using professional expert judgment, would result in inconsistent measures [6].

Since this initial approach, several techniques have been developed to further help decision-makers and financial analysts to better support their decisions by considering traditional statistical learning methods and machine learning and deep learning modelling techniques. Simple linear classification models remain a popular choice among credit institutions, largely due to their adequate accuracy, straightforward implementation, and interpretability of results [7]. The most popular single-period classification techniques are regression models which include, but are not limited to, Linear Discriminant Analysis (LDA) [8] and Logistic Regression Analysis (LR) [9-11]. Empirical results show that the classification accuracy of traditional scoring models is a function of (i) explanatory risk variables, (ii) the training dataset, and (iii) the cut-off point [12]. In the meantime, advancements made in credit risk modelling allowed for the development of newer and more contemporary techniques to assess credit risk amid advances in computer technology. Since credit risk analysis is similar to pattern-recognition problems, machine learning algorithms can be used to classify the creditworthiness of counterparties [13]. Some noteworthy algorithms include Decision Trees (DT) [14], Support Vector Machines (SVM) [15], K-Nearest Neighbours (KNN) [16], and Artificial Neural Networks (ANN) [17,43].

Despite the usage of single classifiers, it is hard to overstate the importance of model uncertainty for economic modelling. The empirical work in economics and social modelling is subject to a large amount of uncertainty about the model specification [18]. The standard approach to risk modelling is to pursue a “winner-take-all” perspective by which, for each dataset, a single believed to be the “best” model is selected from a set of candidate approaches using some method or criteria often neglecting model uncertainty for statistical inference purposes. The use of different lookback periods, diverse selection procedures, alternative accuracy metrics, misspecification problems, and the presence of structural breaks in the data-generating process can lead to different model choices and predictive accuracy [19-24]. When presented with multiple candidate models or algorithms picking one model that can give optimal performance for future data can result in unstable or useless information. Recently, there is a growing interest in homogenous and heterogeneous model combinations in credit risk modelling due to its ability to produce a more robust classifier since it includes the predictions

from all the considered models while circumventing any potential preference [25,26]. The widely used ensemble methods are bagging, boosting, and stacking. Bagging and boosting, who consider homogenous weak learners, combine the results of multiple models to get a generalized result. Bagging (short for bootstrap aggregation) is a parallel procedure; where randomly sampled datasets are produced from the original data taking the average of these predictions to make a final prediction. Boosting is a sequential procedure; it follows a sequential ensemble technique where each subsequent model corrects the error of the earlier model. By identifying these misclassifications in previous iterations, more weight is given to them thus in the next iteration the learner will focus more on these misclassifications.

Traditional model classifications have some limitations. First, the model weights are often calculated using a single validation set. Second, the approach is sensitive to the choice of the model set and the prior distribution of each model. Third, current approaches often use the same weights for all forecasting horizons. Stacked regression ensembles offer an alternative approach to tackling some of the shortcomings of traditional model combination methods. Stacking considers heterogeneous weak learners combining models of diverse types. The architecture of a stacking ensemble involves two or more base models, often referred to as level-0 models with different weight combinations, and a meta-model combining individual models predictions.

This paper empirically investigates the predictive accuracy of single-based classifiers against the stacking generalization approach in credit risk modelling. Four individual classifiers were considered in the analysis as level-0 models and inputs for the meta-model. The model set includes classifiers using machine learning methods (KNN, SVM, and DT) and traditional statistical learning methods (LR). The datasets considered in this paper are provided by Lending Club, a peer-to-peer lending company, and include all accepted and rejected loans from 2007 to 2018. The findings suggest that despite the overall good performance of individual classifiers, the application of ensemble learning techniques can contribute to improving the predictive accuracy of traditional classifiers in credit risk scoring. However, when applying an ensemble model combination, the researcher should pay close attention to these considerations: (i) a reduction of model interpretability when using an ensemble-based approach due to the increase in model complexity, which may make it harder to draw any crucial business insights and (ii) Computation time, an ensemble is a very computationally expensive approach which might not be suitable for real-life applications.

The remainder of the paper is organized as follows. Section 2 summarises the materials and methods used in this study, including the empirical strategy adopted. Section 3 reports the empirical findings of the study. Finally, section 4 critically discusses the results and concludes, pointing directions for future research directions.

2 Materials and methods

2.1 Stacked regression ensemble for credit scoring

Ensembles are sets of learning machines that combine data from different modelling approaches to obtain more dependable and more accurate predictions in supervised and

unsupervised learning problems [27]. An ensemble contains several learners, called based learners, generated by a base learning algorithm. Most ensemble methods produce homogeneous base learners, i.e., learners with the same characteristics, leading to homogeneous ensembles, but some methods utilize multiple learning algorithms to reproduce heterogeneous learners, i.e., learners with distinctive characteristics leading to heterogeneous ensembles techniques [28]. The generalization ability of an ensemble is usually much stronger compared to that of a single classifier [29,44].

Stacking combines point forecasts from multiple heterogeneous base learners using weights that optimize a cross-validation criterion [30]. The base learners can, for example, be alternative statistical learning, machine learning, and deep learning credit scoring methods. The customary stacked regression ensemble approach comprises two modules, base learners (level-0) and a meta-model (level-1), and proceeds in two steps in generating the final credit scoring predictions. The first step (level-0), base learning, consists of multiple base credit risk models which separately generate cross-validated predictions from the training data. The predictions from various models and the observed response variable compose the metadata. In the second step (level-1), a meta-learner is trained on the metadata (low-level output) to estimate the optimal weights for combining multiple base learners while minimizing some optimization (e.g., cross-validation) criterion. The trained meta-classifier is then used to make an overall final prediction by considering each prediction made by the base-level classifier.

For blending the individual forecasts, several choices are possible such as simple averaging, weighted averaging, Bayesian model ensembles, model confidence set, linear or logistic regression, adaptative blending, and neural network (see, e.g., [45-51]). For instance, stacked regression ensemble credit scoring is a linear combination of individual base learners' forecasts in which the model combination weights are coefficients of a linear regression model with observed default rates as the dependent variable and the point forecasts from the individual learners as the independent variables.

2.2 Individual credit scoring models and stacking framework

An ensemble is composed of several base learners. This paper considers four widely used base learners, i.e., LR, SVM, DT, and KNN to investigate the predictive accuracy of stacking in credit risk modelling. The LR [30] is regarded as an industry standard because of its simplicity and balanced error distribution . LR models are fitted via maximum likelihood estimation which involves maximizing a likelihood function to find the probability distribution and parameters that best explain the data. In other words, during the fitting process, this method will estimate the coefficients in such a way it will maximize the probability of labelling an applicant as default as well as maximize the probability of labelling an applicant as non-default.

The SVM [31] algorithm is a state-of-the-art technique with a good generalization performance and computational efficiency. By applying the concept of margin, given a labelled training set, the SVM constructs and uses a discriminant hyperplane or a set of hyperplanes to identify classes. Intuitively, a good separation is achieved by the hyperplane with maximum distance to the nearest training data samples (i.e., maximum margin) therefore an optimal hyperplane maximizes the margin. Both the SVM and LR

algorithms were trained using Stochastic Gradient Descent (SGD), an iterative procedure for optimizing an objective function with suitable smoothing properties.

The DT algorithm is a graphical representation with the primary goal of creating a model that predicts the value of a target variable based on several input variables. The algorithm is based on a measure of data impurity that determines the split of each node, using different metrics. These metrics are applied to each node, and the resulting values are combined (i.e., averaged) to provide a measure of the quality of the split [32]. Then, the impurity of each test is compared, and the split with the lowest impurity is chosen. This process is then continued for each node of the tree to find the "purest" node in terms of the target variable.

The KNN is a distance-based algorithm, taking a majority vote between the "K" closest observations. The main idea of the KNN algorithm is that whenever there is a new point to predict, its nearest neighbours are chosen from the training data [33]. It is easy to understand and implement, with no need to estimate parameters or training. During this phase, the use of machine learning pipelines facilitated the implementation of different transformations such as dealing with missing values, and outliers, encoding categorical variables, and proper scaling. Furthermore, cross-validation and hyperparameter tunning were performed to ensure robustness in the classifiers. Finally, principal component analysis was performed to reduce the number of dimensions in the dataset and discard irrelevant information. For the setup of the stacking ensemble, we considered the LR model as the meta classifier while the remainder classifiers were considered the base learners. Figure 1 schematizes the implementation of the stacking ensemble framework when forecasting the default probability.

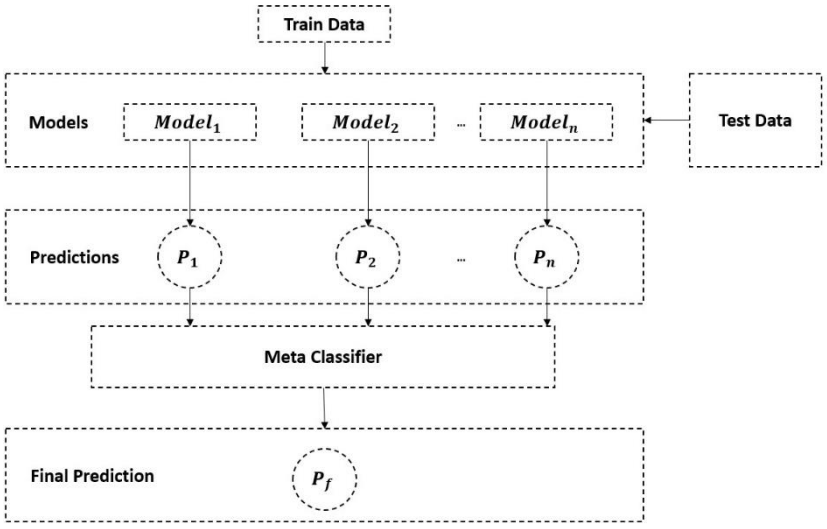

Figure 1. Stacking ensemble framework when forecasting the default probability.
Source: Author's Preparation based on [34]

Four different model combinations have been investigated in this paper. Table 1 provides a synopsis of the ensemble combinations considered. The final estimator is

trained using cross-validated predictions of the base estimators. The meta-learner model used in this classification exercise is set to logistic regression.

Table 1. Model Combinations

Combination	Base Learners	Meta Model
SC1	KNN and SVM	LR
SC2	DT and SVM	LR
SC3	DT and KNN	LR
SC4	DT, KNN, and SVM	LR

Source: Author's preparation.

2.3 Sample credit dataset and variables

The data considered in this paper is provided by Lending Club, a peer-to-peer lending company that matches lenders and borrowers through an online platform without the need for any financial intermediation. The datasets comprise all loan applications issued between 2007 and 2018, including both accepted loans and reject loans, totalling 2.26 million records and 151 features. The dataset is unbalanced, so we follow for instance [35], and use a combination of both undersampling and oversampling combined with stratified K-fold cross-validation to achieve a more balanced distribution.

One of the biggest challenges in building credit scoring models is to select the most relevant features to be used in model calibration. The feature selection process is important to mitigate the overfitting in the curse of dimensionality of classification algorithms and eliminate irrelevant and/or highly correlated variables. This may ultimately lead to a misleading interpretation of the credit scoring model and poor predictive performance. The feature selection process used in this paper includes Principal Component Analysis (PCA) a technique used to eliminate irrelevant, incomplete, and/or unreliable information while also increasing algorithm performance [36]. Out of the 151 features available in the data, the feature selection process identified twenty-three for modelling purposes. These features focus on: (i) Personal details (e.g., annual income, address, and homeownership); (ii) Credit history (e.g., the balance of accounts, revolving and current past due accounts); and (iii) Loan characteristics (e.g., purpose, grade, term). A summary of these characteristics is reported in Table 2.

To calibrate the individual credit scoring models, all structures are trained on the training set comprising 80% of the data, with the remaining 20% used for testing the model's predictive performance. Following the split, additional pre-processing steps were implemented to treat missing values, encode categorical features, and standardize the data for modelling. To achieve this, a pipeline was built using the libraries provided by the python package `sklearn`. The machine learning pipeline procedure automates the workflow it takes to produce a machine learning model from data extraction and pre-processing to model training and deployment. In addition to cross-validation, the tunning of hyperparameters was performed using a parameter grid to maximize the underlying score of the estimator.

Table 2. Chosen features for modelling credit risk

Characteristics	Selected features
Borrower	Annual income
	Homeownership (rent, own, mortgage, other)
	Verification status
	Address State
	Last payment amount.
Credit	Debt-to-Income-Ratio
	Delinquency status in the last 2 years
	Number of inquiries in the last 6 months
	The number of open credit lines in the borrower's credit file
	The number of current open credit lines
	Number of derogatory public records
	Total credit revolving balance
	Revolving line utilization rate
	Number of mortgage accounts
	FICO Score
	The number of public record bankruptcies.
Loan	Loan amount
	Loan term
	Loan interest rate
	LC assigned loan grade
	Purpose

Source: Author's preparation.

The performance obtained in each combination of hyperparameters was measured using the F1-score. Table 3 summarizes the hyperparameters chosen for model optimization and the tuned parameters used for predictions.

Table 3. Hyperparameter optimization

Models	Hyperparameters	Values	Chosen Values
Logistic Regression	Penalty	L1, L2	L2
	Model Alpha	10**-3, 10**-2 , 10**-1	10**-2
Support Vector Machine	Penalty	L1, L2	L1
	Model Alpha	10**-3, 10**-2 , 10**-1	10**-3
K-Nearest Neighbours	Neighbourhood Size	3, 5, 7, 9, 11	9
	Metric	Minkowski	Minkowski
Decision Tree	Criterion	Gini, Entropy	Entropy
	Minimum Sample Splits	20, 30, 50, 100	100
	Maximum Depth	3, 5, 10, 15, 20	15

Source: Author's preparation.

2.4 Predictive performance evaluation criteria

The evaluation criteria for our experiments are adopted from established standard measures in the field of credit scoring [37]. These measures include confusion matrices, the ROC (Receiver Operating Characteristic) curve, and AUC (Area under the ROC Curve) score. A confusion matrix is a table layout that allows the visualization of the performance of an algorithm [38] where the rows represent instances of the actual class, and the columns represent instances of the predicted class. From the confusion matrix, four different metrics can be derived: (i) accuracy, (ii) precision, (iii) recall, and (iv) F1-score. These metrics are computed as follows:

Table 3. Confusion matrix for credit scoring

		Predicted Values	
		Negative = 0	Positive = 1
Actual Values	Negative = 0	True Negative (TN)	False Positive (FP)
	Positive = 1	False Negative (FN)	True Positive (TP)

Source: Author's preparation.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

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

$$F1 - Score = 2 * \frac{(Precision * Recall)}{Precision + Recall} \tag{4}$$

A ROC curve is a graphical plot for visualizing, organizing, and selecting classifiers based on their performance. It shows the relationship between recall and false positive rate (also known as the probability of false alarm) at different threshold points. The AUC score represents the area enclosed by the ROC curve. It is a measure of the ability of a classifier to distinguish between both classes with a value range between 0 and 1. The model fitting, forecasting, simulation procedures, and additional computations have been implemented using a python software routine.

3 Results

The classification algorithms were trained with the selected features. Tables 4 and 5 summarize the predictive accuracy metrics obtained for the individual credit-scoring learners before and after hyperparameter optimization with no sampling techniques applied, respectively.

Table 4. First Experimental Results (Before hyperparameter tunning)

	Accuracy	Precision	Recall	F1-score	AUC Score
LR	0.7438	0.4221	0.7676	0.5446	0.8325
SVM	0.7374	0.4156	0.7766	0.5415	0.7528
KNN	0.8315	0.6012	0.4639	0.5237	0.7845
DT	0.9135	0.7876	0.7757	0.7816	0.5768

Source: Author's preparation.

Table 5. First Experimental Results (After hyperparameter tunning)

	Accuracy	Precision	Recall	F1-score	AUC Score
LR	0.7489	0.4276	0.7607	0.5474	0.8329
SVM	0.7454	0.4238	0.7667	0.5459	0.7564
KNN	0.8396	0.6372	0.4566	0.5320	0.8453
DT	0.9258	0.7492	0.9433	0.8351	0.9764

Source: Author's preparation.

The results from the first experiment suggest that the usage of hyperparameter tuning can increase the overall performance of a classifier. The DT classifier had the best overall performance among the different classifiers. It had the highest F1-score of 0.8351 which is the proportion of charged-off applicants correctly classified and the overall capability of the classifier to predict default. Furthermore, the AUC score of the DT algorithm, which represents the classifier's ability to distinguish classes, outperformed the remaining learners in terms of predictive performance. The LR and the SVM, trained using the SGD algorithm, exhibit similar results with the LR coming slightly ahead with an F1-score of 0.5474 compared to 0.5459, respectively. Regarding the KNN, despite its relatively good performance in the training data, it was evident that the algorithm was overfitting. As training examples increase, the model is not able to perform well on unseen data. It is capturing specific blunders from the data that prevent a good generalization. After this initial analysis, the sampling technique was applied to the dataset. Table 6 summarizes the individual learners' predictive accuracy obtained in the second experiment:

Table 6. Second Experimental Results (Sampling applied)

	Accuracy	Precision	Recall	F1-score	AUC Score
LR	0.8387	0.5636	0.8514	0.6783	0.9185
SVM	0.8379	0.5618	0.8584	0.6782	0.8450
KNN	0.8607	0.6003	0.8829	0.7167	0.9265
DT	0.9262	0.7514	0.9422	0.8361	0.9732

Source: Author's preparation.

The results show an increase in predictive accuracy across all four models. The DT algorithm, despite the loss in its recall score, continues to outperform the other credit scoring models after applying sampling techniques. Interestingly, the DT algorithm

benefited the least from the sampling approach compared to the remaining classifiers. During the implementation of the second experiment, the problem of overfitting that was present in the KNN algorithm vanished, highlighting the importance of considering sampling methods in learning models. Having a set of heterogeneous base-level learners is essential for a successful stacking solution. After performing both hyperparameter optimization and sampling, the best base-level learners and meta-learners were taken to perform the ensemble approach experiments. Table 7 summarizes the model combinations predictive accuracy metrics.

Table 7. Results for each combination

	Accuracy	**Precision**	**Recall**	**F1 Score**	**AUC Score**
SC1	0.8623	0.6072	0.8790	0.7182	0.9335
SC2	0.9265	0.7525	0.9415	0.8365	0.9726
SC3	0.9284	0.7587	0.9404	0.8399	0.9739
SC4	0.9277	0.7564	0.9409	0.8386	0.9731

Source: Author's preparation.

The results in Table 7 suggest the stacked regression ensembles consistently outperform most individual credit scoring models in predicting the default probability in the sample loan data used in this study. The best ensemble is SC3 – stacking with DT and KNN as the base level models followed by SC4, the combination that includes all models. These results suggest that the usage of the SVM algorithm as a base learner in the stacking ensemble may be confusing the meta learner, lowering the importance of base learners' predictions, and decreasing the overall performance of the stacking ensemble. Furthermore, SC1 – stacking with SVM and KNN performed much worse than the single best classifier, the DT algorithm further reinforcing its robustness and predictive capabilities.

This improved predictive accuracy suggests that by assigning model weights that maximize out-of-sample prediction accuracy, the stacking ensemble tends to optimally combine the features of alternative credit scoring models to form a meta-learner credit scoring model, which captures the features of the loan data more adequately than any individual credit risk classifier. The stacking ensemble approach assigns model weights to individual learners that express the out-of-sample performance of the credit scoring classifiers and incorporate future loan data uncertainty into the weights by minimizing the cross-validation criterion. Although the stacking ensemble approaches improvements over the individual base learners, the increased improvement may not always be worth the increased complexity. A stacking ensemble is a good approach for pushing the performance limits, but the improvements of the approach and the costs need to be independently evaluated by the researcher.

4 Conclusions and discussion

Credit scoring methods serve as a risk management tool that allows a financial institution to check applicant reliability to pay off the debt in time. With the developments

made in machine learning applications, credit scoring replaces the reliance on "gut feeling" with statistical analysis reducing the potential risks associated with personal judgment. In this study, a comparative assessment between an ensemble approach and individual classifiers is carried out. Several lessons have been learned which are summarized as follows. The first is that one of the biggest challenges of building robust credit scoring models using real-world data is to select the most relevant credit risk features to be used in the modelling. Data must be properly pre-processed to mitigate overfitting in the curse of dimensionality and eliminate irrelevant and/or highly correlated variables. The results obtained in this study suggest that the feature selection process contributed to improving the performance of individual credit risk models and the super-learner scoring algorithm, helping models to be simpler, more comprehensive, and with lower classification error rates. Secondly, the results suggest that the usage of hyperparameter tuning can increase the overall performance of a credit risk classifier. This result is consistent with previous literature (see, e.g., [39]) and means hyperparameters can impact the algorithm's behaviour by affecting its properties, structure, or complexity, and that the customary use of the default values in the statistical packages adopted to implement super learning raises concerns over the performance of meta-learners. In adopting machine learning methods to credit scoring, hyperparameters should thus be tuned using a grid search procedure whereby the algorithm's cross-validated performance is continually measured over a grid of possible hyperparameter values to optimally identify the best combination. Third, identifying a set of heterogeneous base-level learners is essential for a successful stacking solution. Fourth, the findings suggest the stacked regression ensembles consistently outperform most traditional individual credit scoring models in predicting the default probability. This improved predictive accuracy suggests that the stacking ensemble tends to optimally combine the features of heterogeneous credit scoring models to form a meta-learner credit scoring model which captures the features of the loan application and loan performance data more adequately than any individual classifier.

From a managerial point of view, our findings reinforce the main conclusion that improving credit scoring models to better identify loan delinquency can substantially contribute to reducing loan impairments and losses leading to a significant improvement in the financial performance of credit institutions. Although managers may still be reluctant to implement advanced machine learning and deep learning models to score credit applications due to their higher complexity and the need for qualified experts, the fact that more sophisticated models adopt a data-driven approach to credit decisions, reducing subjective human intervention, judgment and errors may be sufficient to convince them that investing in more robust credit scoring models will ultimately pay-off.

Future research should focus on experimenting with other state-of-the-art machine learning and deep learning model combinations and different sampling techniques such as Synthetic Minority Oversampling Technique (SMOTE) [40]. Unlike random oversampling which only duplicates random instances of the minority class, SMOTE creates synthetic data points that are slightly different from the original data points. These new instances are generated based on the distance of each data point and the minority class's nearest neighbours. This procedure is executed until balancing is achieved. By using SMOTE the possibility of overfitting is minimized and no information is lost

during sampling. However, by introducing these new synthetic points, the possibility of adding noise to the data increases. Additionally, given the challenges that come with unbalanced data, it would be interesting to evaluate a cost-sensitive learning approach since it allows for different misclassification costs for both Type I and Type II errors. Rather than artificially balancing the data distribution via sampling techniques, cost-sensitive learning solves the imbalanced problem by utilizing cost matrices that outline the costs associated with the misclassifications of the various classes [41]. Previous research showed that cost-sensitive learning generates enhanced performance in applications where the dataset has a skewed class distribution [42].

Acknowledgements. This work has been supported by Fundação para a Ciência e a Tecnologia, grants UIDB/04152/2020 - Centro de Investigação em Gestão de Informação (MagIC) and UIDB/00315/2020 (BRU-ISCTE-IUL).

References

1. Ashofteh A., Bravo J. M. (2021). A conservative approach for online credit scoring. Expert System with Applications 176 (2021), Article 114835.
2. Ashofteh A., Bravo J. M. (2019). A non-parametric-based computationally efficient approach for credit scoring. CAPSI 2019 - 19th Conference of the Portuguese Association for Information Systems, Lisbon, Code 160805.
3. Chamboko, R., & Bravo, J. M. (2016). On the modelling of prognosis from delinquency to normal performance on retail consumer loans. Risk Management 18, 264–287.
4. Chamboko, R., & Bravo, J. M. (2020). A Multi-State Approach to Modelling Intermediate Events and Multiple Mortgage Loan Outcomes. Risks, 8(2), 64. http://doi.org/10.3390/risks8020064.
5. Thomas, L.C. (2000). A survey of credit and behavioural scoring: Forecasting financial risk of lending to consumers. International Journal of Forecasting, 16 (2), 149-172.
6. Saunders, A., & Allen, L. (2002). Credit Risk Measurement-New Approaches to Value at Risk and Other Paradigms. John Wiley & Sons, Inc., New York.
7. Lessmann, S., Baesens, B., Seow, H., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. European Journal of Operational Research, 247(1), 124–136.
8. E.I. Altman (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. The Journal of Finance, 23 (4), 589-609.
9. Chamboko, R. & Bravo, J. M. (2019). Modelling and forecasting recurrent recovery events on consumer loans. International Journal of Applied Decision Sciences, 12 (3), 271-287.
10. Chamboko, R. & Bravo, J. M. (2019). Frailty correlated default on retail consumer loans in developing markets. International Journal of Applied Decision Sciences, 12 (3), 257–270.
11. Altman, E.I., Haldeman, R.G. & Narayanan, P. (1977). ZETATM analysis A new model to identify bankruptcy risk of corporations. Journal of Banking and Finance, 1(1), 29-54.
12. Ala'raj, M., & Abbod, M. F. (2016). Classifiers consensus system approach for credit scoring. Knowledge-Based Systems, 104, 89–105.
13. Barboza, F., Kimura, H., & Altman, E. (2017). Machine learning models and bankruptcy prediction. Expert Systems With Applications, 405-417.
14. Zhang, D., Zhou, X., Leung, S., & Zheng, J. (2010). Vertical bagging decision trees model for credit scoring. Expert System with Applications 37, 7838-7843.

15. Huang, C. L., Chen, M. C., & Wang, C. J. (2007). Credit scoring with a data mining approach based on support vector machines. Expert Systems with Applications, 33(4), 847–856.
16. Mukid, M., Widiharih, T., Rusgiyono, A., & Prahutama, A. (2018). Credit scoring analysis using weighted k-nearest neighbour. Journal of Physics: Conference Series 1025, 012114.
17. West, D. (2000). Neural network credit scoring models. Computers and Operations Research, 27(11-12), 1131–1152.
18. Steel, M. F. J. (2020). Model Averaging and Its Use in Economics. Journal of Economic Literature, 58, 644–719.
19. Ashofteh, A., Bravo, J. M., & Ayuso, M. (2022). A New Ensemble Learning Strategy for Panel Time-Series Forecasting with Applications to Tracking Respiratory Disease Excess Mortality during the COVID-19 pandemic. Applied Soft Computing 128, Article 109422.
20. Bravo, J. M., Ayuso, M., Holzmann, R., & Palmer, E. (2021). Addressing the Life Expectancy Gap in Pension Policy. Insurance: Mathematics and Economics, 99, 200-221.
21. Bravo, J. M. (2021). Pricing participating longevity-linked life annuities: A Bayesian Model Ensemble approach. European Actuarial Journal 12, 125–159.
22. Ayuso, M., Bravo, J. M., Holzmann, R., & Palmer, E. (2021). Automatic Indexation of the Pension Age to Life Expectancy: When Policy Design Matters. Risks, 9(5), 96. http://doi.org/10.3390/risks9050096
23. Bravo, J. M., & Ayuso, M. (2020). Mortality and life expectancy forecasts using Bayesian model combinations: An application to the Portuguese population. RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação, E40, 128–144. https://doi.org/10.17013/risti.40.128–145.
24. Bravo, J. M., & Ayuso, M. (2021). Linking Pensions to Life Expectancy: Tackling Conceptual Uncertainty through Bayesian Model Averaging. Mathematics, 9(24): 3307, 1-27.
25. Feng, X., Xiao, Z., Zhong, B., Qiu, J. & Y. Dong (2018). Dynamic ensemble classification for credit scoring using soft probability. Applied Soft Computing Journal, 65, 139-151.
26. Xia, Y., Liu, C., Da, B. & Xie, F. (2018). A novel heterogeneous ensemble credit scoring model based on bstacking approach. Expert Systems with Applications, 93, 182-199.
27. Re, M., & Valentini, G. (2012). Ensemble methods: A review. Advances in Machine Learning and Data Mining for Astronomy. Chapman & Hall, 563-594. https://doi.org/10.1201/B11822-34
28. Zhou, Z. (2012). Ensemble Methods: Foundations and Algorithms. Chapman and Hall. 15-16. https://doi.org/10.1201/b12207
29. Dietterich, T.G. (2000). Ensemble methods in machine learning. Multiple Classifier Systems: First International Workshop, MCS 2000, Lecture Notes in Computer Science. 1-15.
30. Wolpert, D. (1992). Stacked Generalization. Neural Networks. 5. 241-259.
31. Cox, D. R. (1958). The Regression Analysis of Binary Sequences. Journal of the Royal Statistical Society. Series B (Methodological). 20(2), 215-232.
32. Cortes, C., & Vapnik, V. (1995). Support Vector Network. Machine Learning. 20. 273-297.
33. Jijo, B. T., & Abdulazeez, A. M. (2021). Classification based on decision tree algorithm for machine learning. Journal of Applied Science and Technology Trends, 2(01), 20-28.
34. Zhang, Y., Jianxue, W. (2016). K-nearest neighbors and a kernel density estimator for GEFCom2014 probabilistic wind power forecasting. International Journal Forecasting, 32.
35. Jiang, W., Chen, Z., Xiang, Y., Shao, D., Ma, L., & Zhang, J. (2019). Ssem: A novel self-adaptive stacking ensemble model for classification. IEEE Access, 7, 120337–120349.
36. Marqués, A. I., García, V. & Sánchez, J. S. (2013). On the suitability of resampling techniques for the class imbalance problem in credit scoring. Journal of the Operational Research Society, 64(7), 1060-1070.

37. Mishra, S., Sarkar, U., Taraphder, S., Datta, S., Swain, D., & Saikhom, R. et al. (2017). Multivariate Statistical Data Analysis- Principal Component Analysis (PCA). International Journal of Livestock Research, 7(5), 60-78.
38. Abdou, H., Pointon, J. (2011). Credit Scoring, Statistical Techniques and Evaluation Criteria: A Review of the Literature. Int. Syst. Accounting, Finance and Management 18: 59-88.
39. Powers, D. M. W. (2011). Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. Journal of Machine Learning Technologies 2:37-63.
40. Luo G. (2016). A review of automatic selection methods for machine learning algorithms and hyper-parameter values. Network Modeling Analysis in Health Informatics and Bioinformatics 5:18.
41. Chawla, N., Bowyer, K., Hall, L., & Kegelmeyer, W. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research (JAIR), 16, 321-357.
42. Mienye, D., & Sun, Y. (2021). Performance analysis of cost-sensitive learning methods with application to imbalanced medical data. Informatics in Medicine Unlocked, 25, 1-10.
43. Yu, H., Sun, C., Yang, X., Zheng, S., Wang, Q. & Xi, X. (2018). LW-ELM: A Fast and Flexible Cost-Sensitive Learning Framework for Classifying Imbalanced Data. IEEE Access 6, 28488-28500.
44. Ampountolas, A., Nyarko Nde, T., Date, P., & Constantinescu, C. (2021). A Machine Learning Approach for Micro-Credit Scoring. Risks, 9(3), 50.
45. Bravo, J. M., & Ayuso, M. (2021) Forecasting the Retirement Age: A Bayesian Model Ensemble Approach. In: Rocha Á., Adeli H., Dzemyda G., Moreira F., Ramalho Correia A.M. (eds) Trends and Applications in Information Systems and Technologies. WorldCIST 2021. Advances in Intelligent Systems and Computing, Volume 1365 AIST, pp. 123 – 135. Springer, Cham. https://doi.org/10.1007/978-3-030-72657-7_12
46. Bravo, J. M., & Ayuso, M. (2021) Forecasting the Retirement Age: A Bayesian Model Ensemble Approach. In: Rocha Á. et al. (eds) Trends and Applications in Information Systems and Technologies. WorldCIST 2021. Advances in Intelligent Systems and Computing, Vol 1365 AIST, pp. 123 – 135. Springer, Cham. https://doi.org/10.1007/978-3-030-72657-7_12
47. Ashofteh, A. & Bravo, J. M. (2021). Life Table Forecasting in COVID-19 Times: An Ensemble Learning Approach. In A. Rocha, R. Gonçalves, F. G. Penalvo, & J. Martins (Eds.), Proceedings of CISTI 2021 - Iberian Conference on Information Systems and Technologies. IEEE Computer Society Press. https://doi.org/10.23919/CISTI52073.2021.9476583.
48. Bravo, J. M., El Mekkaoui, N. (2022). Short-Term CPI Inflation Forecasting: Probing with Model Combinations. In: Rocha, A. et al. (eds) Information Systems and Technologies. WorldCIST 2022. Lecture Notes in Networks and Systems, vol 468. Springer, Cham, pp. 564–578. https://doi.org/10.1007/978-3-031-04826-5_56.
49. Bravo, J. M., Ayuso, M. (2020). Mortality and life expectancy forecasts using Bayesian model combinations: An application to the portuguese population. RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao, E40, 128–144 (Dec 2020). https://doi.org/10.17013/risti.40.128–145.
50. Ashofteh, A., Bravo, J. M. & Ayuso, M. (2021). A Novel Layered Learning Approach for Forecasting Respiratory Disease Excess Mortality during the COVID-19 pandemic. CAPSI 2021 Proceedings, Volume 2021 – October 2021, Code 183080.
51. Bravo, J. M. (2020). Longevity-Linked Life Annuities: A Bayesian Model Ensemble Pricing Approach. CAPSI 2020 Proceedings. 29. https://aisel.aisnet.org/capsi2020/29 (Atas da 20ª Conferência da Associação Portuguesa de Sistemas de Informação 2020).
52. Bouttier, F., and H. Marchal, 2020: Probabilistic thunderstorm forecasting by blending multiple ensembles. Tellus A, 72 (1), 1–19.