Machine Learning for Credit Scoring: Improving Logistic Regression with Non Linear Decision Tree Effects

Elena Ivona Dumitrescu, Sullivan Hué, Christophe Hurlin, Sessi Tokpavi

Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects ☆

Elena Dumitrescu [a], Sullivan Hué [b,*], Christophe Hurlin [b], Sessi Tokpavi [b]

[a] EconomiX-CNRS, University of Paris Nanterre, 200 Avenue de la République, Nanterre 92000, France
[b] Univ. Orléans, CNRS, LEO (FRE 2014), Rue de Blois, Orléans 45067, France

A B S T R A C T

In the context of credit scoring, ensemble methods based on decision trees, such as the random for- est method, provide better classification performance than standard logistic regression models. However, logistic regression remains the benchmark in the credit risk industry mainly because the lack of inter- pretability of ensemble methods is incompatible with the requirements of financial regulators. In this paper, we propose a high-performance and interpretable credit scoring method called penalised logistic tree regression (PLTR), which uses information from decision trees to improve the performance of logistic regression. Formally, rules extracted from various short-depth decision trees built with original predic- tive variables are used as predictors in a penalised logistic regression model. PLTR allows us to capture non-linear effects that can arise in credit scoring data while preserving the intrinsic interpretability of the logistic regression model. Monte Carlo simulations and empirical applications using four real credit default datasets show that PLTR predicts credit risk significantly more accurately than logistic regression and compares competitively to the random forest method.

Keywords: Risk management Credit scoring Machine learning Interpretability Econometrics,

1. Introduction

Credit scoring was one of the first fields of application of machine learning techniques in economics. Some early examples are decision trees (Coffman, 1986; Makowski, 1985; Srinivasan & Kim, 1987), *k*-nearest neighbours (Henley & Hand, 1996; 1997), neural networks (NN) (Desai, Crook, & Overstreet Jr, 1996; Tam & Kiang, 1992; West, 2000; Yobas, Crook, & Ross, 2000), and support vector machines (SVMs) (Baesens et al., 2003). At that time, the accuracy gains (compared to the standard logistic regression model) for creditworthiness assessment appeared to be limited (see the early surveys of Thomas (2000) and Baesens et al. (2003)). However, the performance of machine learning-based scoring models has been improved substantially since the adoption of ensemble (aggregation) methods, especially bagging and boosting methods (Finlay, 2011; Lessmann, Baesens, Seow, & Thomas, 2015; Paleologo, Elisseeff, & Antonini, 2010).[1] In their extensive benchmarking study, Lessmann et al. (2015) compared 41 algorithms with various assessment criteria and several credit scoring datasets. They confirmed that the random forest method, i.e., the randomised version of bagged decision trees (Breiman, 2001), largely outperforms logistic regression and has progressively become one of the standard models in the credit scoring industry (Grennepois, Alvirescu, & Bombail, 2018). Over the last decade, machine learning techniques have been increasingly used by banks and fintechs as challenger models (ACPR, 2020) or sometimes for credit production, generally associated with "new" data (social or communication networks, digital footprint, etc.) and/or "big data" (Hurlin & Pérignon, 2019).[2]

☆ This paper has previously circulated under the title "Machine Learning or Econometrics for Credit Scoring: Let's Get the Best of Both Worlds". We are grateful to Emanuele Borgonovo (the editor), three anonymous referees, Jérémy Leymarie, Thomas Raffinot, Benjamin Peeters, Alexandre Girard, and Yannick Lucotte. We also thank the seminar participants at University of Orléans as well as the participants of the 16th Conference "Développements Récents de l'Econométrie Appliquée à la Finance" (Université Paris Nanterre), 7th PhD Student Conference in International Macroeconomics and Financial Econometrics, 35th Annual Conference of the French Finance Association and International Association for Applied Econometrics for their comments. Finally, we thank the ANR programs MultiRisk (ANR-16-CE26-0015-01), CaliBank (ANR-19-CE26-0002-02), and the Chair ACPR/Risk Foundation: Regulation and Systemic Risk for supporting our research.

* Corresponding author.
E-mail addresses: elena.dumitrescu@parisnanterre.fr (E. Dumitrescu), sullivan.hue@univ-orleans.fr (S. Hué), christophe.hurlin@univ-orleans.fr (C. Hurlin), sessi.tokpavi@univ-orleans.fr (S. Tokpavi).

[1] The ensemble or aggregation methods aim to improve the predictive performance of a given statistical or machine learning algorithm (weak learner) by using a linear combination (through averaging or majority vote) of predictions from many variants of this algorithm rather than a single prediction.

[2] See Óskarsdóttir, Bravo, Sarraute, Vanthienen, & Baesens (2019) or Frost, Gambacorta, Huang, Shin, & Zbinden (2019) for a general discussion about the value of

However, one of the main limitations of machine learning methods in the credit scoring industry comes from their lack of explainability and interpretability. Most of these algorithms, in particular ensemble methods, are considered as "black boxes" in the sense that the corresponding scorecards and credit approval process cannot be easily explained to customers and regulators. This is consistent with financial regulators' current concerns about the governance of AI and the need for interpretability, especially in the credit scoring industry. See, for instance, the recent reports on this topic published by the French regulatory supervisor (ACPR, 2020), the Bank of England (Bracke, Datta, Jung, & Sen, 2019), the European Commission (EC, 2020), and the European Banking Authority (EBA, 2020), among many others. This explains why the logistic regression remains the standard approach in credit industry, due to its simplicity and intrinsic interpretability. Most international banks still use the logistic regression model, especially for regulatory scores used to estimate the probability of default for capital requirements (Basel III) or for point-in-time estimates of expected credit losses (IFRS9).

Within this context, we propose a hybrid credit scoring approach called the *penalised logistic tree regression* model (hereafter PLTR). PLTR aims to improve the predictive performance of the logistic regression model through data pre-processing and feature engineering based on short-depth decision trees and a penalised estimation method while preserving the intrinsic interpretability of the scoring model. Formally, PLTR consists of a simple logistic regression model including predictors extracted from decision trees. These predictors are binary rules (leaves) outputted by short-depth decision trees built with original predictive variables. To handle a possibly large number of decision-tree rules, we incorporate variable selection in the estimation through an adaptive lasso logistic regression model (Friedman, Hastie, & Tibshirani, 2010; Zou, 2006), i.e., a penalised version of classic logistic regression.

The PLTR model has several advantages. First, it allows us to capture non-linear effects (i.e., thresholds and interactions between the features) that can arise in credit scoring data. It is recognised that ensemble methods consistently outperform logistic regression because the latter fails to fit these non-linear effects. For instance, the random forest method benefits from the recursive partitioning underlying decision trees and hence, by design, accommodates unobserved univariate and multivariate threshold effects. The notable aspect of our approach consists of using these algorithms to pre-treat the predictors instead of modelling the default probability directly with machine learning classification algorithms. Second, PLTR provides parsimonious and interpretable scoring rules (e.g., marginal effects or scorecards) as recommended by the regulators, since it preserves the intrinsic interpretability of the logistic regression model and is based on a simple feature selection method.

In this article, we propose several Monte Carlo experiments to illustrate the inability of standard parametric models, i.e., standard logistic regression models with linear specification of the index or with quadratic and interaction terms, to capture well the non-linear effects (thresholds and interactions) that can arise in credit scoring data. Furthermore, these simulations allow us to evaluate the relative performance of PLTR in the presence of non-linear effects while controlling for the number of predictors. We show that PLTR outperforms standard logistic regression in terms of out-of-sample forecasting accuracy. Moreover, it compares competitively to the random forest method while providing an interpretable scoring function. We apply PLTR and six other benchmark credit scoring methodologies (random forest, linear logistic regression, non-linear logistic regression, non-linear logistic regression with adaptive lasso, an SVM and an NN) on four real credit scoring datasets. The empirical results confirm those obtained through simulations, as PLTR yields good forecasting performance for all the datasets. This conclusion is robust to the various predictive accuracy indicators considered by Lessmann et al. (2015) and to several diagnostic tests. Finally, we show that PLTR also leads to more cost reductions than alternative credit scoring models.

Our paper contributes to the literature on credit scoring on various issues. First, our approach avoids the traditional trade-off between interpretability and forecasting performance. We propose here to restrict the intrinsic complexity of credit-score models rather than apply *ex post* interpretability methods to analyse the scoring model after training. Indeed, many model-agnostic methods have been recently proposed to make the "black box" machine learning models explainable and/or their decisions interpretable (see Molnar, 2019 for an overview). We can cite here among many others the partial dependence (PDP) or individual conditional expectation (ICE) plots, global or local (such as the LIME) surrogate models, feature interaction, Shapley values, Shapley additive explanations (SHAPE), etc. In the context of credit scoring models, Bracke et al. (2019) and Grennepois & Robin (2019) promoted the use of Shapley values.[3] Bussman, Giudici, Marinelli, & Papenbrock (2019) recently proposed an explainable machine learning model specifically designed for credit risk management. Their model applies similarity networks to Shapley values so that the predictions are grouped according to the similarity in the underlying explanatory variables. However, obtaining the Shapley values requires considerable computing time because the number of coalitions grows exponentially with the number of predictive variables, and computational shortcuts provide only approximate and unstable solutions. An alternative approach is the InTrees method proposed by Deng (2019). That algorithm extracts, measures, prunes, selects, and summarises rules from a tree ensemble and calculates frequent variable interactions. This helps detect simple decision rules from the forest that are important in predicting the outcome variable. Nevertheless, the algorithms underlying the extraction of these rules are not easy to disclose. Finally, our contribution can also be related to the methods designed to enable NNs and SVMs to be interpretable, especially the TREPAN (Thomas, Crook, & Edelman, 2017), Re-RX (Setiono, Baesens, & Mues, 2008), or ALBA (Martens, Baesens, & Van Gestel, 2008) algorithms. However, there is a slight difference between these approaches and ours. While the latter aim to enable a model (i.e., NNs or SVMs) to be explainable/interpretable, PLTR aims to improve the predictive performance of a simple model (i.e., the logistic regression model) that is inherently interpretable.

Second, our approach can be viewed as a systematisation of common practices in the credit industry, where standard logistic regression is still the standard scoring model, especially for regulatory purposes. Indeed, credit risk modellers usually introduce non-linear effects in logistic regression by using ad hoc or heuristic pre-treatments and feature engineering methods (Hurlin & Pérignon, 2019) such as discretisation of continuous variables, merger of categories, and identification of non-linear effects with cross-product variables. In contrast, we propose here a systematic and automatic approach for modelling such unobserved non-linear effects based on short-depth decision trees. Thus, PLTR may allow model developers to significantly reduce the time spent on data management and data pre-processing steps.

More generally, our paper complements the literature devoted to hybrid classification algorithms. The PLTR model differs from

big data for credit scoring. In the present article, we limit ourselves to the use of machine learning algorithms with "traditional data" for credit risk analysis.

[3] This method assumes that each feature of an individual is a player in a game where its predictive abilities determine the pay-out of each feature (Lundberg & Lee, 2017).

the so-called logit-tree models, i.e., trees that contain logistic regressions at the leaf nodes such as the logistic tree with unbiased selection (LOTUS) in Chan & Loh (2004) and the logistic model tree (LMT) in Landwehr, Hall, & Frank (2005). Although similar in spirit, our PLTR method also contrasts with the hybrid CART-logit model of Cardell & Steinberg (1998). Indeed, to introduce multivariate threshold effects in logistic regression, Cardell & Steinberg (1998) used a single non-pruned decision tree. However, the large depth of this unique tree complicates interpretability and may lead to predictor inflation that is not controlled for (e.g., through penalisation, as in our case). PLTR also shares similarities with the two-step classification algorithm recently proposed by De Caigny, Coussement, & De Bock (2018) in the context of customer churn prediction. Their initial analysis consisted of applying a decision tree to split customers into homogeneous segments corresponding to the leaves of the decision tree, while the second step consisted of estimating a logistic regression model for each segment. However, their method is based on a single non-pruned decision tree as in the hybrid CART-logit model. Furthermore, their objective was to improve the predictive performance of the logistic regression by identifying homogeneous subsets of customers and not by introducing non-linear effects as in the PLTR approach.

The rest of the article is structured as follows. Section 2 analyses the performance of logistic regression and random forest in the presence of univariate and multivariate threshold effects through Monte Carlo simulations. In Section 3, we introduce the PLTR credit scoring method and assess through Monte Carlo simulations its accuracy and interpretability (parsimony) in the presence of threshold effects. Section 4 describes an empirical application with a benchmark dataset. The robustness of the results to dataset choice is explored in Section 5. Section 6 compares the models from an economic point of view, while the last section concludes the paper.

2. Threshold effects in logistic regression

2.1. Non-linear effects and the logistic regression model

Let (x_i, y_i), $i = 1, \ldots, n$ be a sample of size n of independent and identically distributed observations, where $x_i \in \mathbb{R}^p$ is a p-dimensional vector of predictors and $y_i \in \{0, 1\}$ is a binary variable taking the value one when the i-th borrower defaults and zero otherwise. The goal of a credit scoring model is to provide an estimate of the posterior probability $\Pr(y_i = 1|x_i)$ that borrower i defaults given its attributes x_i. The relevant characteristics of the borrower vary according to its status: household or company. For corporate credit risk scoring, the candidate predictive variables $x_{i,j}$, $j = 1, \ldots, p$, may include balance-sheet financial variables that cover various aspects of the financial strength of the firm, such as the firm's operational performance, its liquidity, and capital structure (Altman, 1968).[4] For retail loans, financial variables such as the number and amount of personal loans, normal repayment frequency of loans, the number of credit cards, the average overdue duration of credit cards and the amount of housing loans are combined with socio-demographic factors. A typical example is the FICO score, which is widely used in the US financial industry to assess the creditworthiness of individual customers.

Regardless of the type of borrower, the conditional probability of default is generally modelled using a logistic regression with the following specifications:

$$\Pr(y_i = 1|x_i) = F(\eta(x_i; \beta)) = \frac{1}{1 + \exp(-\eta(x_i; \beta))}, \tag{1}$$

where $F(.)$ is the logistic cumulative distribution function and $\eta(x_i; \beta)$ is the so-called index function defined as

$$\eta(x_i; \beta) = \beta_0 + \sum_{j=1}^{p} \beta_j x_{i,j},$$

where $\beta = (\beta_0, \beta_1, \ldots, \beta_p) \in \mathbb{R}^{p+1}$ is an unknown vector of parameters. The estimator $\widehat{\beta}$ is obtained by maximizing the log-likelihood function

$$\mathcal{L}(y_i; \beta) = \sum_{i=1}^{n} \Big\{ y_i \log\{F(\eta(x_i; \beta))\} + (1 - y_i) \log\{1 - F(\eta(x_i; \beta))\} \Big\}.$$

The main advantage of the logistic regression model is its simple interpretation. Indeed, this model searches for a single linear decision boundary in the predictors' space. The core assumption for finding this boundary is that the index $\eta(x_i; \beta)$ is linearly related to the predictive variables. In this framework, it is easy to evaluate the relative contribution of each predictor to the probability of default. This is achieved by computing marginal effects as

$$\frac{\partial \Pr(y_i = 1|x_i)}{\partial x_{i,j}} = \beta_j \frac{\exp(\eta(x_i; \beta))}{[1 + \exp(\eta(x_i; \beta))]^2},$$

with estimates obtained by replacing β with $\widehat{\beta}$. Thus, a predictive variable with a positive (negative) significant coefficient has a positive (negative) impact on the borrower's default probability.

This simplicity comes at a cost when significant non-linear relationships exist between the default indicator, y_i, and the predictive variables, x_i. A very common type of non-linearity can arise from the existence of a univariate threshold effect on a single predictive variable, but it can also be generalised to a combination of such effects (multivariate threshold effects) across variables. A typical example of the former case in the context of credit scoring is the income "threshold effect", which implies the existence of an endogenous income threshold below (above) which default probability is more (less) prominent. The income threshold effect can obviously interact with other threshold effects, leading to highly non-linear multivariate threshold effects. The common practice to approximate non-linear effects in credit scoring applications is to introduce quadratic and interaction terms in the index function $\eta(x_i; \beta)$. However, such a practice is not successful when unobserved threshold effects are at stake.

To illustrate the inability of standard parametric models, i.e., standard logistic regression model or logistic regression with quadratic and interaction terms, to capture accurately the non-linear effects (thresholds and interactions) that can arise in credit scoring data, we propose a Monte Carlo simulation experiment. In a first step (simulation step), we generate p predictive variables $x_{i,j}$, $j = 1, \ldots, p$, $i = 1, \ldots, n$, where the sample size is set to $n = 5,000$. Each predictive variable $x_{i,j}$ is assumed to follow the standard Gaussian distribution. The index function $\eta(x_i; \Theta)$ is simulated as follows:

$$\eta(x_i; \Theta) = \beta_0 + \sum_{j=1}^{p} \beta_j \mathbf{1}(x_{i,j} \le \gamma_j) + \sum_{j=1}^{p-1} \sum_{k=j+1}^{p} \beta_{j,k} \mathbf{1}(x_{i,j} \le \delta_j) \mathbf{1}(x_{i,k} \le \delta_k), \tag{2}$$

where $\mathbf{1}(.)$ is the indicator function and $\Theta = (\beta_0, \beta_1, \ldots, \beta_p, \beta_{1,2}, \ldots, \beta_{p-1,p})'$ is the vector of parameters, with each component randomly drawn from a uniform $[-1, 1]$ distribution, and $(\gamma_1, \ldots, \gamma_p, \delta_1, \ldots, \delta_p)'$ are some threshold parameters, whose values are randomly selected from the support

[4] For instance, using a sample of 4,796 Belgian firms, Bauweraerts (2016) shows the importance of taking into account the level of liquidity, solvency and profitability of the firm in forecasting its bankruptcy risk. For small and medium enterprises (SMEs), specific variables related to the financial strength of the firm's owner are also shown to be important (Wang, 2012).

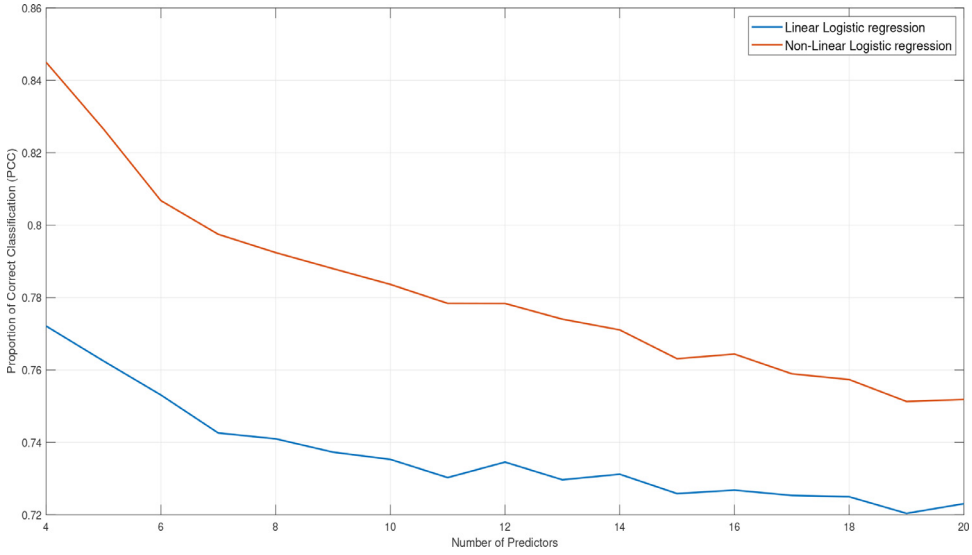

Fig. 1. Comparison of performances under univariate and bivariate threshold effects: linear and non-linear logistic regressions.

of each predictive variable generated while excluding data below (above) the first (last) decile. The default probability is then obtained for each individual by plugging (2) into (1). Subsequently, the simulated target binary variable y_i is obtained as

$$y_i = \begin{cases} 1 & \text{if } \Pr(y_i = 1|x_i) > \pi \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where π stands for the median value of the generated probabilities.

In a second step (estimation step), we estimate by maximum likelihood two logistic regressions on the simulated data $\{y_i, x_i\}_{i=1}^n$: (i) a standard logistic regression model and (ii) a (non-linear) logistic regression with quadratic and interaction terms. For the standard logistic regression model, the conditional probability is based on a linear index defined as

$$\eta(x_i; \beta) = \beta_0 + \sum_{j=1}^{p} \beta_j x_{i,j}.$$

For non-linear logistic regression, we also include quadratic and interaction terms

$$\eta^{(nl)}\left(x_i; \Theta^{(nl)}\right) = \alpha_0 + \sum_{j=1}^{p} \alpha_j x_{i,j} + \sum_{j=1}^{p} \xi_j x_{i,j}^2 + \sum_{j=1}^{p-1} \sum_{k=j+1}^{p} \zeta_{j,k} x_{i,j} x_{i,k}.$$

where $\Theta^{(nl)} = \left(\alpha_0, \alpha_1, \ldots, \alpha_p, \xi_1, \ldots, \xi_p, \zeta_{1,2}, \ldots, \zeta_{p-1,p}\right)'$ is the unknown vector of parameters.

Fig. 1 displays the average value of the percent of correct classification (PCC) values of these two models over 100 simulations and for different numbers of predictors $p = 4, \ldots, 20$.[5] We observe that their accuracy decreases with the number of predictors. This result arises because the two logistic regression models are misspecified because they do not control for threshold and interaction effects, and their degree of misspecification increases with additional predictors. Indeed, in our DGP (i.e., Eq. (2)), the number of regressors controls for the degree of non-linearity of the data generating process: more predictors correspond to more threshold and interaction effects. These results suggest that in the presence of univariate and bivariate threshold effects involving many variables, logistic regression with a linear index function, eventually augmented with quadratic and interaction terms, fails to discriminate between good and bad loans. In the case where $p = 20$, the PCCs of the logistic regression models are equal to only 72.30% and 75.19%.

2.2. Machine learning for non-linear effects

In the context of credit scoring, ensemble methods based on decision trees, such as the random forest method, provide better classification performance than standard logistic regression models (Finlay, 2011; Lessmann et al., 2015; Paleologo et al., 2010). The out-performance of the random forest method arises from the non-linear "if-then-else" rules underlying decision trees.[6] Formally, for a given tree, l, the algorithm proceeds as follows. Let $\mathcal{D}_{m,l}$ be the data (sub)set at a given node m of this tree. We denote by $\theta_{m,l} = \left(j_{m,l}, t_{m,l,j}\right)$ a candidate split, where $j_{m,l} = 1, \ldots, p$ indicates a given predictor and $t_{m,l,j}$ is a threshold value in the support of this variable. The algorithm partitions the data $\mathcal{D}_{m,l}$ into two subsets $\mathcal{D}_{m,l,1}\left(\theta_{m,l}\right)$ and $\mathcal{D}_{m,l,2}\left(\theta_{m,l}\right)$, with[7]

$$\mathcal{D}_{m,l,1}\left(\theta_{m,l}\right) = (x_i, y_i) | x_{i,j} < t_{m,l,j},$$

$$\mathcal{D}_{m,l,2}\left(\theta_{m,l}\right) = (x_i, y_i) | x_{i,j} \geq t_{m,l,j},$$

where the parameter estimates $\widehat{\theta}_{m,l}$ satisfy

$$\widehat{\theta}_{m,l} = (\widehat{j}_{m,l}, \widehat{t}_{m,l,j}) = \arg\max_{\theta_{m,l}} \mathcal{H}\left(\mathcal{D}_{m,l}\right) - \frac{1}{2}\Big(\mathcal{H}\left(\mathcal{D}_{m,l,1}\left(\theta_{m,l}\right)\right) + \mathcal{H}\left(\mathcal{D}_{m,l,2}\left(\theta_{m,l}\right)\right)\Big),$$

with $\mathcal{H}(.)$ a measure of diversity, e.g., the Gini criterion, applied to the full sample and averaged across the two sub-samples. Hence,

[5] We divide the simulated sample into two sub-samples of equal size at each replication. The training sample is used to estimate the parameters of the logistic regression model, while the classification performance is evaluated on the test sample. To compute the PCC, we estimate y_i by comparing the estimated probabilities of default, $\widehat{p}_i$, to an endogenous threshold $\widehat{\pi}$. As usual in the literature, we set $\widehat{\pi}$ to a value such that the number of predicted defaults in the learning sample is equal to the observed number of defaults.

[6] Indeed, the latter is a non-parametric supervised learning method based on a divide-and-conquer greedy algorithm that recursively partitions the training sample into smaller subsets to group together as accurately as possible individuals with the same behaviour, i.e., the same value of the binary target variable "y_i".

[7] To simplify the description of the algorithm, we focus only on quantitative predictors. A similar procedure is available for qualitative predictors.

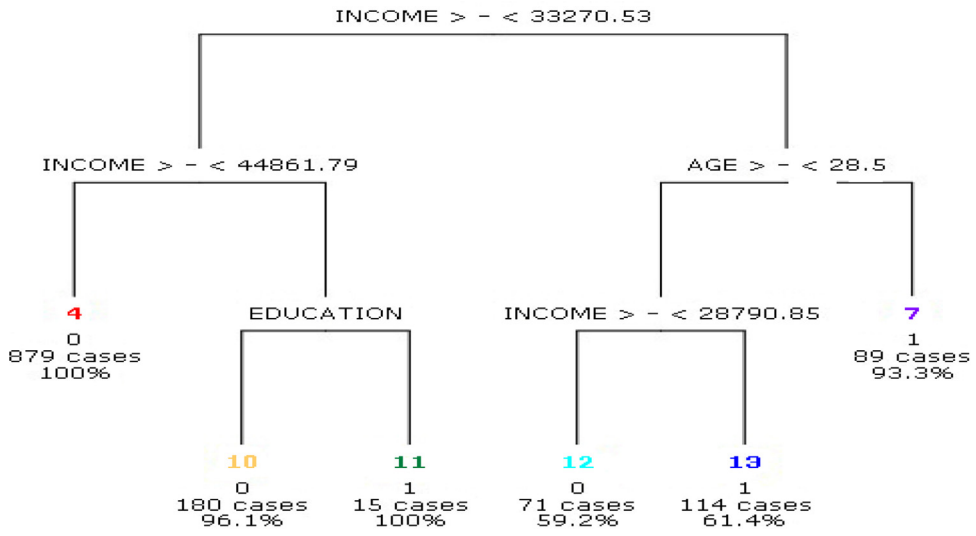

Fig. 2. Example of a decision tree for credit scoring.

$\widehat{\theta}_{m,l}$ appears as the value of $\theta_{m,l}$ that reduces diversity the most within each subset resulting from the split. The splitting process is repeated until the terminal sub-samples, also known as leaf nodes, contain homogeneous individuals according to a predefined homogeneity rule. We denote by M_l the total number of splits in tree l and by $|T_l|$ the corresponding number of leaf nodes.

An illustrative example of a decision tree is given in Fig. 2. At the first iteration (or split), $m = 1$, $\widehat{\theta}_{m,l}$ is defined by $(\widehat{j}_{m,l}, \widehat{t}_{m,l,1})$, with $\widehat{j}_{m,l}$ the index of the variable "income" and $\widehat{t}_{m,l,1} = 33270.53$. The other iterations also include "age" and "education" for further refinements. The process ends with a total number of 5 splits and 6 leaf nodes labelled 10, 11, 12, 13, 4 and 7. Each leaf node $\mathcal{R}_t$, $t = 1, \ldots, |T_l|$ includes a specific proportion of individuals belonging to each class of borrowers (1="default", 0="no default"). For instance, leaf node "7" contains 89 individuals, 93.3% of them having experienced a default event. Note that each of these individuals has an income lower than 33270.53 and is less than 28.5 years old. The predominant class in each leaf defines the predicted value of y_i for individuals i that belong to that particular leaf. Formally, the predicted default value for the ith individual is

$$h_l(x_i; \widehat{\Theta}_l) = \sum_{t=1}^{|T_l|} c_t \mathcal{R}_{i,t},$$

where $\Theta_l = \left(\theta_{m,l}, m = 1, \ldots, M_l\right)$ is the parameter vector for tree l, $\mathcal{R}_{i,t} = 1_{(i \in \mathcal{R}_t)}$ indicates whether individual i belongs to leaf $\mathcal{R}_t$, and c_t is the dominant class of borrowers in that leaf node. For example, in leaf node 7, the "default" class is dominant; hence, the predicted value $h_l(x_i)$ is equal to 1 for all the individuals that belong to this leaf node. Note that this simple tree allows us to identify both interaction and threshold effects. For instance, in the simple example of Fig. 2, the predicted value can be viewed as the result of a kind of linear regression[8] on the product of two binary variables that take a value of one if the income is lower than 33270.53 and the age is less than 28.5.

The random forest method is a bagging procedure that aggregates many uncorrelated decision trees. It exploits decision-tree power to detect univariate and multivariate threshold effects while reducing their instability. Its superior predictive performance springs from the variance reduction effect of bootstrap aggregation for non-correlated predictions (Breiman, 1996). Let L trees be constructed from bootstrap samples (with replacement) of fixed size drawn from the original sample. To ensure a low level of correlation among those trees, the random forest algorithm chooses the candidate variable for each split in every tree, $j_{m,l}$ with $m \in \{1, \ldots, M_l\}$ and $l \in \{1, \ldots, L\}$, from a restricted number, p_l, of randomly selected predictors among the p available ones, with $p_l = \lceil \sqrt{p} \rceil$. The default prediction of the random forest for each borrower, $h(x_i)$, is obtained by the principle of majority vote; that is, $h(x_i)$ corresponds to the mode of the empirical distribution of $h_l(x_i; \widehat{\Theta}_l)$, $l = 1, \ldots, L$.

To illustrate the ability of the random forest method to capture the non-linear effects that can arise in credit scoring data well, we consider the same Monte Carlo framework as in Section 2.1. The proportion of correct classification for the random forest algorithm, displayed as a yellow line in Fig. 3, is computed over the same test samples of length 2500 as the PCCs of the logistic regressions previously discussed. The optimal number of trees in the forest, L, is tuned using the out-of-bag error. Our results confirm the empirical findings of the literature: in the presence of non-linear effects, random forest outperforms not only linear logistic regression (as expected) but also non-linear logistic regression. This illustrates the ability of random forests to capture both threshold and interaction effects between the predictors well. These findings are valid regardless of the number of predictors, even if the differences in classification performance between the three models are decreasing in the number of predictors.[9] Indeed, as the number of predictors increases, the complexity and the non-linearity of the DGP also increases, which diminishes the performance of all the classifiers. For instance, the PPCs are equal to 99.18% (resp. 84.50%) for the random forest (resp. logistic regression with quadratic and interaction terms) in the case with 4 predictors, against 81.20% (resp. 75.19%) in the case with 20 predictors.

Despite ensuring good performance, the aggregation rule (majority vote) underlying the random forest method leads to a prediction rule that lacks interpretation. This opacity is harmful for credit scoring applications, where decision makers and regulators need simple and interpretable scores (see ACPR (2020) and EC (2020),

[8] This equivalence is true only in the case of a regression tree when the target variable y is continuous.

[9] See Vapnik & Chervonenkis (1971) for a theoretical result on the fact that theoretical risk or generalisation error for any machine learning algorithm decreases with the number of observations and increases with the complexity given by the number of predictors, or more generally the so-called VC dimension.

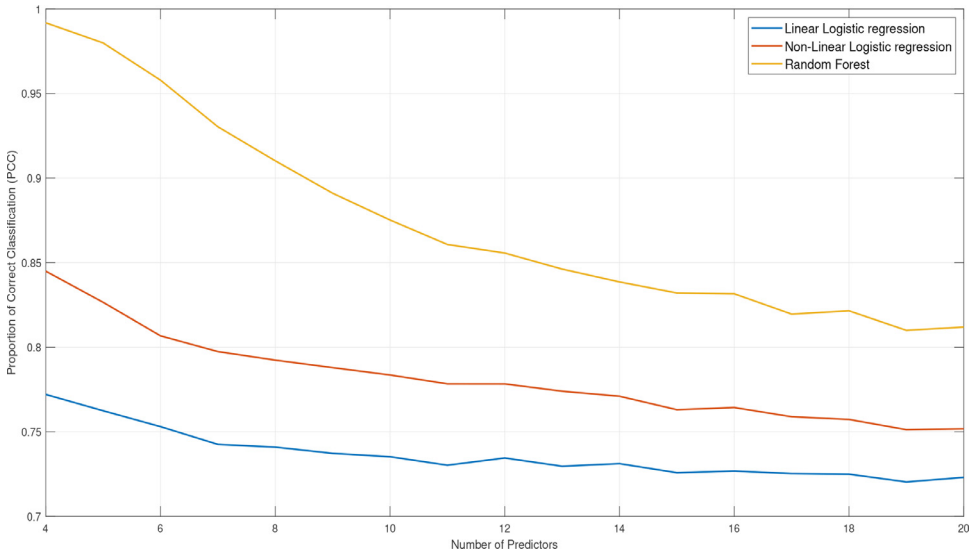

Fig. 3. Comparison of performances under univariate and bivariate threshold effects: linear and non-linear logistic regressions and the random forest method.

among many others). The key question here is how to find a suitable trade-off between predictive performance and interpretability. To address this issue, two lines of research can be explored. First, one can try to diminish the complexity of the random forest method's aggregation rule by selecting (via an objective criterion) only some trees or decision rules in the forest.[10] Second, we can preserve the simplicity of logistic regression while improving its predictive performance with univariate and bivariate endogenous threshold effects. We opt here for the second line of research, with the PLTR hybrid scoring approach.

3. Penalised logistic tree regression

3.1. Description of the methodology

PLTR aims to improve the predictive performance of the logistic regression model through new predictors based on short-depth decision trees and a penalised estimation method while preserving the intrinsic interpretability of the scoring model. The algorithm proceeds in two steps.

The objective of the first step is to identify threshold effects from trees with one and two splits. The case of one-split trees is simple: for each explanatory variable in the analysis, regardless of its level of informativity, two multicolinear leaf nodes are created and one of them is retained for the subsequent analysis. Overall, we count p such threshold effects, one for each predictive variable. To understand more easily the case of two-split trees, let us take an example (see Fig. 4). Consider the income and age as the jth and kth explanatory variables, respectively, and assume that income is more informative than age in explaining credit default. For each individual i, the corresponding decision tree generates three binary variables, each associated with a terminal node. The first binary variable $\mathcal{V}_{i,1}^{(j)}$ accounts for univariate threshold effects and takes the value of one when the income of individual i is higher than an estimated income threshold and zero otherwise. The second (third) binary variable $\mathcal{V}_{i,2}^{(j,k)}$ ($\mathcal{V}_{i,3}^{(j,k)}$), representing bivariate threshold effects, is equal to one when the person's income is lower than its threshold and at the same time his/her age is higher (lower) than an estimated age threshold and zero otherwise.[11] Note that this particular form of splitting should arise when both variables are informative, i.e., each of them is selected in the iterative process of splitting. If the second variable is non-informative (age), the tree relies twice on the first informative variable (income).

One leaf of the longer branch of the tree is retained so as to cover two-splits threshold-effects, i.e., $\mathcal{V}_{i,2}^{(j,k)}$ in this example. We count at most q such threshold effects, $q \leq p \times (p-1)/2$, corresponding to the total number of couples of predictors. We could also retain the leaf coming from the first split of the tree, $\mathcal{V}_{i,1}^{(j)}$, but this would come down to a subset of the p threshold effects identified with one-split trees. Indeed, the most informative predictive variables may be selected in the first split of several two-split trees, while the others, less informative, may never be retained as the most relevant predictor. Consequently, they do not produce any $\mathcal{V}_{i,1}^{(j)}$ in this setup even though they are relevant on their own, i.e. in one-split trees. To avoid this loss of information, we regroup the p univariate threshold effects from one-split trees and the q bivariate threshold effects corresponding to the $\mathcal{V}_{i,2}^{(j,k)}$ leaf of two-split trees. Still, some bivariate threshold effects may be redundant. Indeed, redundancy can arise from the fact that some variables may be selected in the splits of several trees. Consequently, we remove the redundant threshold effects before moving to the following step.[12]

In the second step, the endogenous univariate and bivariate threshold effects previously obtained are plugged in the logistic regression

$$\Pr\left(y_i = 1 | \mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta\right) = \frac{1}{1 + \exp\left[-\eta(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta)\right]}, \qquad (4)$$

[10] Note that this is the approach underlying the so-called InTrees method of Deng (2019), who proposed a methodology to render the random forest outputs interpretable by extracting simple rules from a tree ensemble.

[11] It is also possible that the univariate threshold variable $\mathcal{V}_{i,1}^{(j)}$ takes the value of one when the income is lower than an estimated income threshold, and zero otherwise. In that case, the bivariate threshold effect $\mathcal{V}_{i,2}^{(j,k)}$ ($\mathcal{V}_{i,3}^{(j,k)}$) is equal to one when the individual's income is higher than its threshold and at the same time his/her age is higher (lower) than an estimated age threshold, and zero otherwise.

[12] Note that one could also go beyond two splits by analyzing triplets or quadruplets of predictive variables. Such a procedure would allow the inclusion of more complex non-linear relationships in the logistic regression. Nevertheless, the expected uprise in performance would come at the cost of increased complexity of the model toward that of random forests, which would plunge its level of interpretability. For this reason, in our PLTR model, we use only short-depth decision trees involving one and two splits.

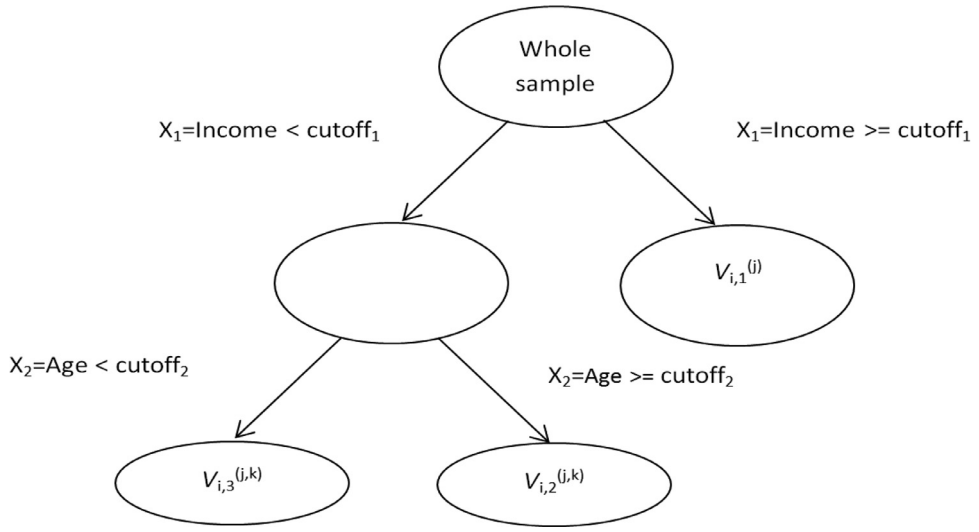

Fig. 4. Illustration of the two-stage splitting process.

with

$$\eta(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta) = \beta_0 + \sum_{j=1}^{p} \alpha_j x_i + \sum_{j=1}^{p} \beta_j \mathcal{V}_{i,1}^{(j)} + \sum_{j=1}^{p-1} \sum_{k=j+1}^{p} \gamma_{j,k} \mathcal{V}_{i,2}^{(j,k)}$$

the index and $\Theta = (\beta_0, \alpha_1, \ldots, \alpha_p, \beta_1, \ldots, \beta_p, \gamma_{1,2}, \ldots, \gamma_{p-1,p})'$ the set of parameters to be estimated. The corresponding log-likelihood is

$$\mathcal{L}(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta) = \frac{1}{n} \sum_{i=1}^{n} \left[y_i \log \left[F\left(\eta(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta) \right) \right] + (1 - y_i) \log \left[1 - F\left(\eta(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta) \right) \right] \right],$$

where $F(\eta(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta))$ is the logistic cdf. The estimate $\widehat{\Theta}$ is obtained by maximizing the above log-likelihood with respect to the unknown parameters Θ. Note that the length of Θ depends on the number of predictive variables, p, which can be relatively high. For instance, there are 45 couples of variables when $p = 10$; this leads to a maximum number of 55 univariate and bivariate threshold effects that play the role of predictors in our logistic regression.

To prevent overfitting issues in this context with a large number of predictors, a common approach is to rely on penalisation (regularisation) for both estimation and variable selection. In our case, this method consists of adding a penalty term to the negative value of the log-likelihood function, such that

$$\mathcal{L}_p(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta) = -\mathcal{L}(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta) + \lambda P(\Theta), \tag{5}$$

where $P(\Theta)$ is the additional penalty term and λ is a tuning parameter that controls the intensity of the regularisation and which is selected in such a way that the resulting model minimises the out-of-sample error. The optimal value of the tuning parameter λ is usually obtained by relying on a grid search with cross-validation or by using some information criteria. In addition, several penalty terms $P(\Theta)$ have been proposed in the related literature (Tibshirani, 1996; Zou, 2006; Zou & Hastie, 2005). Here, we consider the adaptive lasso estimator of Zou (2006). Note that the adaptive lasso satisfies the oracle property; i.e., the probability of excluding relevant variables and selecting irrelevant variables is zero, contrary to the standard lasso penalisation (Fan & Li, 2001). The corresponding penalty term is $P(\Theta) = \sum_{v=1}^{V} w_v |\theta_v|$ with $w_v = |\widehat{\theta}_v^{(0)}|^{-\nu}$, where $\widehat{\theta}_v^{(0)}$, $v = 1, \ldots, V$, are consistent initial estimators of the parameters and ν is a positive constant. The adaptive lasso estimators are obtained as

$$\widehat{\Theta}_{alasso}(\lambda) = \arg\min_{\Theta} - \mathcal{L}\left(\mathcal{V}_{i,1}^{(j)}, \mathcal{V}_{i,2}^{(j,k)}; \Theta \right) + \lambda \sum_{v=1}^{V} w_v |\theta_v|. \tag{6}$$

In practice, we set the parameter ν to 1 and the initial estimator $\widehat{\theta}_j^{(0)}$ to the value obtained from the logistic-ridge regression (Hoerl & Kennard, 1970), and the only free tuning parameter, λ, is found via 10-fold cross-validation.[13] Note also that since decision tree algorithms are immune to collinearity by nature and the adaptive lasso regression is consistent in variable selection for Generalized Linear Models even when the irrepresentable condition is violated, the PLTR method is robust to collinearity issues.

Instead of penalizing the logistic regression to select only the most relevant threshold effects, we could have selected only the binary variables (the leaves of the trees) corresponding to the trees with the highest predictive power.[14] However, two main issues arise from this approach. Firstly, measuring the performance of the short-depth decision trees over the training sample is a potential source of overfitting. A natural solution would be to divide the current training sample into a pure training sample, and a validation sample to measure the out-of-sample performances of the trees, but this strategy is reliable only if the initial dataset is particularly large. Secondly, implementing a pre-selection procedure supposes setting a threshold γ on the scale of predictive performance, beyond (below) which a decision tree and therefore its leafs will be retained (excluded). Setting this threshold is challenging, as too high values could exclude relevant trees and too low values could include irrelevant ones. In contrast, our approach, which consists in retaining all non-redundant binary variables outputted by all the decision trees and in selecting the most relevant ones in the second step (within the penalized logistic regression) does not suffer from these shortcomings. As the adaptive Lasso has the oracle property, it will select only binary variables related to trees with

[13] Different estimation algorithms have been developed in the literature to estimate regression models with the adaptive lasso penalty (for a given value of λ): the quadratic programming technique (Shewchuk et al., 1994), the shooting algorithm (Zhang & Lu, 2007), the coordinate-descent algorithm (Friedman et al., 2010), and the Fisher scoring algorithm (Park & Hastie, 2007). Most of them are implemented in software such as MATLAB and R, and we rely here on the algorithm based on Fisher scoring. See McIlhagga (2016) for more details on this optimisation algorithm.

[14] We thank an anonymous referee for this suggestion.

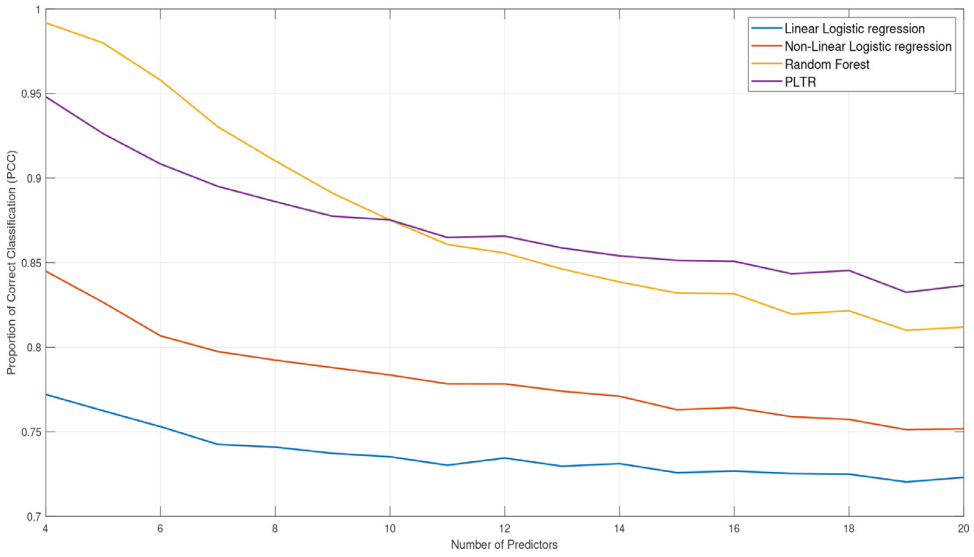

Fig. 5. Comparison of performances under univariate and bivariate threshold effects: linear and non-linear logistic regressions, the random forest method and PLTR.

high predictive power. More precisely, the calibrated penalty parameter λ in the adaptive Lasso is positively related to the unknown γ threshold.

In summary, PLTR is a hybrid classification model designed to increase the predictive power of the logistic regression model via feature engineering. Its first step consists of creating additional binary predictors based on short-depth decision trees built with singletons and couples of predictive variables. These binary variables are then introduced, in a second step, in a penalised logistic regression model, where the adaptive lasso is used for both estimation and variable selection.

3.2. PLTR under threshold effects: Monte Carlo evidence

In this subsection, we assess the accuracy and interpretability of the PLTR method in the presence of threshold effects. For that, we consider the same Monte Carlo experiment as that defined in Section 2.

Fig. 5 displays the PCC for our PLTR method computed over the same test samples of length 2,500 that were generated with the DGP in (2)-(3). The main conclusion is that the PLTR method outperforms the two versions of the logistic regression, i.e., with and without quadratic and interaction terms. Equally important, when there are few predictors, i.e., p is small, the PCC of PLTR is lower than that of random forest. However, as p increases, the performance of PLTR approaches that of the random forest method, and both models have approximately the same classification performance. For example, the PCCs are equal to 94.81 for our new method and 99.18 for the random forest with $p = 4$, against 83.65 and 81.20 for $p = 20$, respectively. Note that the latter case seems more realistic, as credit scoring applications generally rely on a large set of predictors in practice.

We gauge the robustness of these findings to the choice of evaluation criteria by computing on the same test samples two other performance measures for all models under analysis, namely the Area Under the ROC Curve (AUC) and the Brier Score (BS).[15] The results, displayed in Figures C.1 and C.2 in Section C of the online appendix, are very similar to those obtained for the PCC, and the conclusions remain the same. The random forest and PLTR outperform both versions of the logistic regression and the performance of each model decreases with the complexity and non-linearity of the DGP. Moreover, the PLTR compares competitively to the random forest and even surpasses it as the number of predictors increases.

Performance is not the only essential criterion for credit scoring managers. The other fundamental characteristic of a good scoring model is interpretability.[16] Interpretability and accuracy are generally two competing objectives: the first is favoured by simple models, the latter by complex models. Moreover, the degree of interpretability of a credit scoring model is difficult to measure. As discussed in Molnar (2019), there is no real consensus in the literature about what is interpretable for machine learning, nor is it clear how to measure this factor. Doshi-Velez & Kim (2017) distinguishes three levels of evaluation of interpretability: the application level, the human level, and the function level. While the application and human levels are related to the understanding of the conclusion of a model (from an expert or a layperson, respectively), the function level corresponds to the evaluation of decision rules from a statistical viewpoint (for example, the depth of a decision tree). In the specific context of credit scoring, Bracke et al. (2019) distinguishes six different types of stakeholders (developers, 1st- and 2nd-line model checkers, management, regulators, etc.).[17] Each of them has its own definition of what interpretability should

[15] The Area Under the ROC Curve and Brier Score are complementary performance measures to the PCC and related to different facets of the predictive performance of scorecards, i.e., the accuracy of the scores and the discriminatory power of the classifiers. We provide a complete description of these performance measures in Section 4.2.

[16] Computation time may also be a relevant criterion. We compute the average time required to run each classifier over 100 simulations over the same test samples previously considered. All computation times are very reasonable for the problem at hand. It increases with the complexity of the approach: logistic regression are almost instantaneous, followed closely by the random forest and the PLTR. Notice that most of the computation time of the PLTR model comes from the penalisation, which is standard in the literature and insures a proper selection of the relevant predictors, and not from the identification of threshold effects, which render PLTR superior to the standard logistic regression. We are grateful to an anonymous referee for this suggestion.

[17] Bracke et al. (2019) distinguished the (i) developers, i.e., those developing or implementing an ML application; (ii) 1st-line model checkers, i.e., those directly responsible for ensuring that model development is of sufficient quality; (iii) management responsible for the application; (iv) 2nd-line model checkers, i.e., staff that, as part of a firm's control functions, independently check the quality of model development and deployment; (v) conduct regulators that are interested in deployed

be and how to measure it. For instance, the developer and 1st-line checkers may be interested in individual predictions when they obtain customer queries and in better understanding outliers. In contrast, second-line model checkers, management, and prudential regulators are likely to adopt a more general viewpoint and may be less interested in individual predictions.

In the credit scoring context, interpretability can be measured from at least two perspectives. First, one can consider simple metrics such as the size of the set of decision rules. This indicator allows us to compare models in terms of ease of interpretation: the fewer the rules in a decision set, the easier it is for a user to understand all the conditions that correspond to a particular class label. The size of a given rule in a decision set is a complementary measure. Indeed, if the number of predicates in a rule is too large, it will lose its natural interpretability. This perspective corresponds to the function level evaluation mentioned by Doshi-Velez & Kim (2017). Second, one can interpret the decision rules through marginal effects, elasticities, or scorecards. This second perspective corresponds to the human-level evaluation evoked by Doshi-Velez & Kim (2017) or to the global model interpretability defined by Molnar (2019). Which features are important and what kind of interactions take place between them?

We confirm this trade-off between interpretability and classification performance. The less accurate model, i.e., the logistic regression model, is intrinsically interpretable through marginal effects or explicit scorecard. In contrast, the model with the highest classification performance among our competing models, i.e., the random forest model, is not interpretable for two reasons. First, the forest relies on many trees with many splits, which involves many complicated if-then-else rules. Second, the rules obtained from the trees are aggregated via the majority vote.

Within this context, our PLTR method is a parsimonious solution to the trade-off between performance and interpretability. The scoring decision rules are simple to interpret through marginal effects (as well as elasticities and scorecards) similar to those of traditional logistic regression. This is facilitated by the simple decision rules obtained in the first step of the procedure from short-depth decision trees. Indeed, the skeleton of our PLTR is actually a logistic regression model with binary indicators that account for endogenous univariate and bivariate threshold effects. The complete loan-decision process based on the PLTR method is illustrated in Fig. 6. The input of the method includes all the predictive variables from the loan applicant, while the output is fundamentally the decision to accept or to reject the credit application based on the default risk of the person. Additionally, the mapping from the inputs to the output allows one to transform the internal set of rules of PLTR into transparent feedback about the weaknesses and strengths of the application.

To provide more insights into interpretability, we compare our PLTR model and the random forest in the same Monte Carlo setup as in Section 2, with p fixed to 20, using simple metrics. We consider the two metrics previously defined, i.e., the size of the set of decision rules and the size of a given rule in the decision set. Across the 100 simulations, the random forest registers an average number of 160.9 trees, each with an average number of 410.5 terminal nodes. This leads to a decision set of 410.5×160.9 binary decision variables or rules that can be used for prediction with this method. Across the same simulations, the average number of active binary decision variables in our penalised logistic regression is equal to 146.9.[18] Moreover, the number of predicates involved in each of these binary decision variables for our PLTR method varies between 1 and 2 by construction, whereas the maximum number of predicates in a rule of the random forest model is 14.5 on average. Hence, the PLTR model appears to be easier to interpret than the random forest model and comparable to non-linear logistic regression in this sense.[19]

Furthermore, marginal effects and elasticities can be easily obtained in PLTR due to the linearity of the link function (cf. Eq. (4)). On the one hand, this greatly simplifies significance testing as well as the implementation of out-of-sample exercises. On the other hand, this allows credit institutions to easily explain, in a transparent way, the main reasons behind a loan decision.

4. Model performance with a benchmark dataset

As a complement to Monte Carlo simulations, we now consider an empirical application based on a benchmark credit default dataset to assess the practical usefulness of PLTR.

4.1. Data description and processing

To gauge the out-of-sample performance of the PLTR method and to illustrate its interpretability, we use a popular dataset provided by a financial institution for the Kaggle competition "Give me some credit", which is often used in credit scoring applications (Baesens et al., 2003). The dataset includes several predictive variables and a binary response variable measuring default. The predictive variables provide information about the customers (age, monthly income, the number of dependents in the family) and the application form (number of mortgage and real estate loans, the monthly debt payments, the total balance on credit cards, etc.). The dataset contains 10 quantitative predictors. See Table A.1 in Section A of the online appendix for a description of the variables in the dataset.

The number of instances in the dataset is equal to 150,000 loans out of which 10,026 defaults, leading to a prior default rate of 0.067.[20] All the missing values have been replaced by the mean of the predictive variable. Finally, regarding data partitioning, we use the so-called $N \times 2$-fold cross-validation of Dietterich (1998), which involves randomly dividing the dataset into two subsamples of equal size. The first (second) part is used to build the model, while the second (first) part is used for evaluation. This procedure is repeated N times, and the evaluation metrics are averaged. This method of evaluation produces more robust results compared to classical single data partitioning. We set $N = 5$ for computational reasons.

4.2. Statistical measures of performance and interpretability

To evaluate the performance of each classifier, we use five accuracy measures considered by Lessmann et al. (2015) in their benchmarking study: the area under the ROC curve (AUC), the Brier score (BS), the Kolmogorov-Smirnov statistic (KS), the percentage of correctly classified (PCC) cases, and the partial Gini index (PGI). These indicators are related to different facets of the predictive performance of scorecards, namely, the accuracy of the scores as measured by the BS statistics, the quality of the classification given by

models being in line with conduct rules and (vi) prudential regulators that are interested in deployed models being in line with prudential requirements.

[18] Note that for $p = 20$ predictors, the maximum number of binary variables is equal to $20 + \frac{20 \times 19}{2} = 210$. This result illustrates the selection processed through adaptive lasso regression.

[19] The major difference between these two methods is the endogenous character of the thresholds that characterise variable interactions in our framework.

[20] It is well known that imbalanced classes impede classification: some classifiers may focus too much on the majority class and neglect the minority group (of interest). They can hence exhibit good overall performance despite poorly identifying the minority group, i.e., the borrowers that default. A common solution consists of using an under-sampling or over-sampling method, such as SMOTE. Nonetheless, here, we choose not to resample the data, as the prior default rate is larger than 6%.

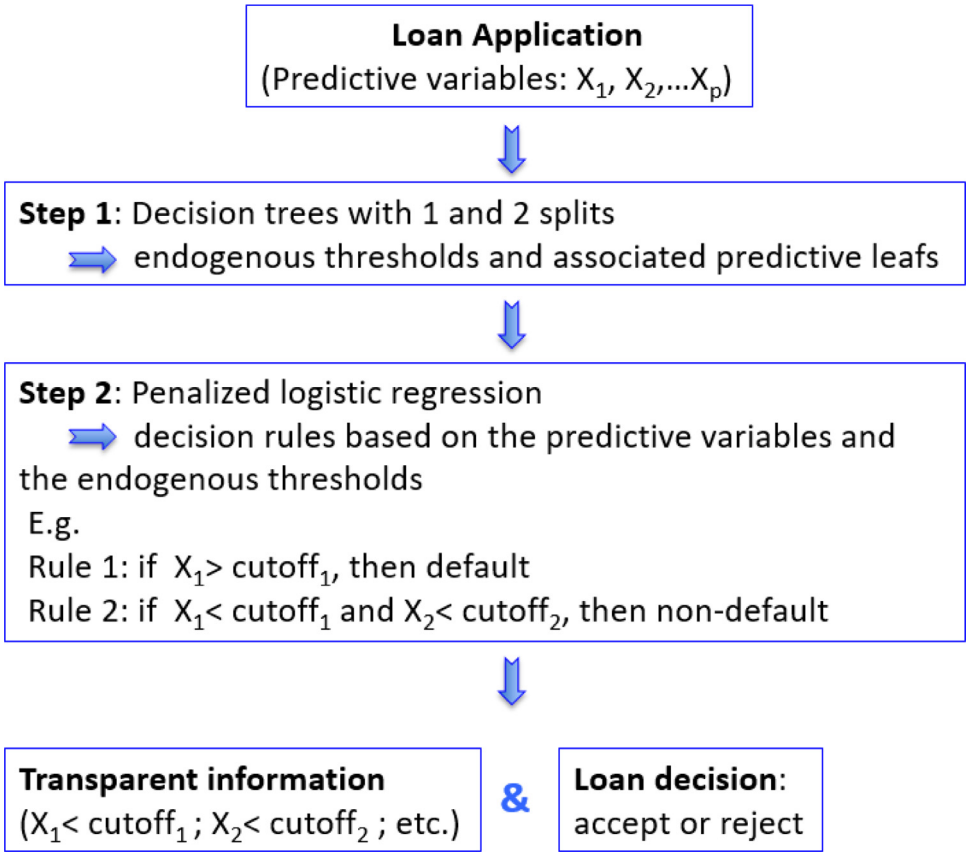

Fig. 6. PLTR inference process.

the PCC and KS statistics, and the discriminatory power assessed through the AUC and the PGI statistics. By using several statistics instead of a single one, we expect to obtain a robust and complete evaluation of the relative performances of the competing models.

The AUC tool evaluates the overall discriminatory performance of each model or classifier. It is a measure of the link between the false positive rate (FPR) and the true positive rate (TPR), each computed for every threshold between 0 and 1. The FPR (TPR) is the percentage of non-defaulted (defaulted) loans misclassified (correctly classified) as defaulted. Thus, AUC reflects the probability that the occurrence of a randomly chosen bad loan is higher than the occurrence of a randomly chosen good loan.

The Gini index is equal to twice the area between the ROC curve and the diagonal. Hence, similar to the AUC metric, it evaluates the discriminatory power of a classifier across several thresholds, with values close to one corresponding to perfect classifications. However, in credit scoring applications, it is not realistic to study all possible thresholds. Informative thresholds are those located in the lower tail of the distribution of default probabilities (Hand, 2005). Indeed, only applications below a threshold in the lower tail can be granted a credit, which excludes high thresholds. The partial Gini index solves this issue by focusing on thresholds in the lower tail (Pundir & Seshadri, 2012). With x denoting a given threshold and $L(x)$ denoting the function describing the ROC curve, the PGI is then defined as[21]

$$PGI = \frac{2\int_a^b L(x)dx}{(a+b)(b-a)} - 1.$$

[21] The PGI within bounds $a = 0$ and $b = 1$ is equivalent to the Gini Index. In the empirical applications, we evaluate the PGI within the $(0, 0.4)$ bounds as in Lessmann et al. (2015).

The PCC is the proportion of loans that are correctly classified by the model. Its computation requires discretisation of the continuous variable of estimated probabilities of default. Formally, we need to choose a threshold π above (below) which a loan is classified as bad (good). In practice, the threshold π is fixed by comparing the costs of rejecting good customers/granting credits to bad customers. Since we do not have such information, we set this threshold to a value such that the predicted number of defaults in the learning sample is equal to the observed number of defaults.

The Kolmogorov-Smirnov statistic is defined as the maximum distance between the estimated cumulative distribution functions of two random variables. In credit scoring applications, these two random variables measure the scores of good loans and bad loans (Thomas, Edelman, & Crook, 2002).

Lastly, the Brier score (Brier, 1950) is defined as

$$BS = \frac{1}{n}\sum_{i=1}^{n}(\widehat{\Pr}(y_i = 1|x_i) - y_i)^2,$$

where $\widehat{\Pr}(y_i = 1|x_i)$ is the estimated probability of default and y_i is the target binary default variable. Note that it is the equivalent of the mean-square error but designed for the case of discrete-choice models. Overall, the higher these indicators are, the better the model is, except for the Brier score, for which a smaller value is better.

Regarding the interpretability of the scoring models, the criteria retained to compare PLTR and the random forest method are the size of the decision set and the average size of rules in a decision set (see also Section 3.2).

Table 1
Statistical performance indicators: Kaggle dataset.

Methods	AUC	PGI	PCC	KS	BS
Linear Logistic Regression	0.6983	0.3964	0.9082	0.3168	0.0576
Non-Linear Logistic Regression	0.7660	0.5255	0.9127	0.4173	0.0649
Non-Linear Logistic Regression + ALasso	0.8062	0.6102	0.9208	0.4751	0.0535
Random Forest	0.8529	0.6990	0.9260	0.5563	0.0500
PLTR	0.8568	0.7076	0.9247	0.5647	0.0496
Support Vector Machine	0.7418	0.4830	0.9117	0.3723	0.0619
Neural Network	0.7517	0.5006	0.9074	0.3895	0.0552

Note: Non-linear logistic regression includes linear, quadratic and interaction terms. The method labelled "Non-Linear Logistic Regression + ALasso" corresponds to a penalised version of non-linear logistic regression with an adaptive lasso penalty.

4.3. Statistical evaluation results

Table 1 presents the average value of each statistic across the 5×2 cross-validation test samples. We compare the out-of-sample performance of PLTR to that of traditional logistic regression and the random forest method. Three different versions of the logistic regression are implemented: simple linear logistic regression, its non-linear version, which includes as additional variables quadratic and interaction terms,[22] and a penalised version of this last model to avoid overfitting due to the large number of predictors. We use the adaptive lasso penalty as described above. These augmented logistic regression models are used to assess the importance of non-linear effects of the features. We also include an SVM and NN in the comparison, as they are widely used for credit scoring applications in the literature (Baesens et al., 2003; Lessmann et al., 2015; Thomas, 2000).

The results displayed in Table 1 show that the random forest method performs better than the three versions of the logistic regression, and this holds for all statistical measures considered. In particular, the differences are more pronounced for the AUC, PGI and KS statistics. Our PLTR method also performs better than the three versions of logistic regression irrespective of the performance measure. This is particularly applicable for the AUC, PGI and KS metrics, for which the dominance is stronger. The take-away message here is that combining decision trees with a standard model such as logistic regression provides a valuable statistical modelling solution for credit scoring. In other words, the non-linearity captured by univariate and bivariate threshold effects obtained from short-depth decision trees can improve the out-of-sample performance of traditional logistic regression. The SVM and NN results are consistent with those in the literature (Baesens et al., 2003; Grennepois et al., 2018; Lessmann et al., 2015; Thomas, 2000). They are slightly better than those of the logistic regression model, but these methods generally perform less well than ensemble learning methods such as the random forest method. Most importantly, these models also perform less well than PLTR.

The results in Table 1 also show that PLTR compares competitively to the random forest method. All statistical performance measures are of the same order. Therefore, the two methods exhibit similar statistical performance, and neither of them should be preferred over the other based on these criteria. However, the parsimony of PLTR contrasts with the complexity underlying the prediction rule of the random forest method. To illustrate this point, Table 2 displays the interpretability measures for the random forest method and PLTR, as well as that of linear logistic regression for comparison purposes. The average number of trees in the random forest method across the 5×2 cross-validation test samples is equal to 173.9. These trees have on average 5,571.1 terminal nodes, with a total of $5,571.1 \times 173.9$ binary variables for prediction (via the majority vote). By contrast, the average number of bivariate threshold effects selected by our penalised logistic regression is only 40. More importantly, these bivariate threshold effects are easily interpretable because they arise from short-depth decision trees. In addition, the PLTR rules are built from only 2 predicates at most, whereas the rules from the random forest method are built from an average number of 32.2 predicates at most. Overall, both criteria confirm that PLTR is easier to interpret than the random forest method. These differences in terms of the size of the decision set and size of the rules between both models are the penalty of capturing more non-linear effects, although such effects do not seem to play a significant role in this dataset. For comparison, the average number of predictors is 11 for linear logistic regression, each of them relying on a single predicate. The PLTR results are not very different from those of linear logistic regression, with the gap corresponding to the non-linear effects included in our model to improve the performance of the benchmark linear logistic regression method.

Lastly, to highlight the interpretability advantage of our method, we report in Table 3 the 10 most important decision rules from short-depth decision trees, which are selected by an adaptive lasso in the implementation of our PLTR method. These decision rules are associated with the largest absolute values of the marginal effects (averaged across individuals). A positive (negative) value of a given marginal effect provides information about the strength of an increase (decrease) of the probability of default. We observe that three univariate threshold variables are selected, i.e., "NumberOfTime60-89DaysPastDueNotWorse <0.5", "NumberOfTimes90DaysLate<0.5" and "RevolvingUtilizationOfUnsecuredLines<0.69814", the first one appearing as the most important in terms of marginal effect. Referring to the description of this variable in Table A.1 of the online appendix, we can infer that the probability of default is 3.92% less important when the number of times a borrower has been between 60 and 89 days past due (but not worse in the last 2 years) is lower than 0.5 compared to the reference case when this number is higher than 0.5. Moreover, seven bivariate threshold effects are selected by the models as being important in explaining credit default. This kind of analysis that helps measure through marginal effects the importance of the decision rules from the short-depth decision trees is an important added value of our PLTR model in terms of interpretability.

5. Robustness across datasets

In this section, we evaluate the out-of-sample robustness of the above empirical results across datasets. To this end, we consider three popular additional datasets. The first one, named "Housing", is available in an SAS library and has been used by many authors for illustrative examples (Matignon, 2007). The second dataset, labelled the "Australian dataset", concerns credit card applications and is a University of California at Irvine (UCI) dataset provided

[22] As already stressed, this non-linear model is the one that is generally used to capture non-linear effects in the framework of logistic regression.

Table 2
Measures of interpretability: Kaggle dataset.

Methods	Size of the decision set	Maximal number of predicates
Linear Logistic Regression	11	1
Random Forest	5,571.1 × 173.9	32.2
PLTR	40	2

Note: This table displays the average values of interpretability measures for linear logistic regression, the random forest method and PLTR.

Table 3
Decision rules and average marginal effects: full sample Kaggle dataset.

#	Decision rules	Average marginal effects
1	"NumberOfTime60-89DaysPastDueNotWorse <0.5"	-0.0392
2	"NumberOfTimes90DaysLate<0.5" & "RevolvingUtilizationOfUnsecuredLines<0.59907"	-0.0389
3	"NumberOfTimes90DaysLate<0.5" & "NumberOfTime60-89DaysPastDueNotWorse<0.5"	-0.0342
4	"NumberOfTime60-89DaysPastDueNotWorse<0.5" & "NumberOfTime30-59DaysPastDueNotWorse<0.5"	-0.0326
5	"NumberOfTimes90DaysLate<0.5"	-0.0326
6	"NumberOfTime60-89DaysPastDueNotWorse>=0.5" & "NumberOfTime60-89DaysPastDueNotWorse<1.5"	-0.0300
7	"RevolvingUtilizationOfUnsecuredLines>=0.69814" & "RevolvingUtilizationOfUnsecuredLines<1.001"	-0.0285
8	"RevolvingUtilizationOfUnsecuredLines<0.69814"	-0.0281
9	"NumberOfTimes90DaysLate<0.5" & "NumberOfTime30-59DaysPastDueNotWorse<0.5"	-0.0277
10	"NumberOfTimes90DaysLate<0.5" & "NumberOfTime30-59DaysPastDueNotWorse<0.5"	-0.0231

Note: The table provides the list of the decision rules associated with the 10 largest absolute values of the marginal effects (with respect to the probability of defaulting) derived from the PLTR model estimated using the full sample. See Table A.1 in the online appendix for a precise description of the variables.

by Quinlan, and it was used as a credit approval database in the Statlog project.[23] Lastly, the third dataset, labelled the "Taiwan dataset", is also a UCI dataset that collects information about default payments in Taiwan.

The Housing dataset includes 5960 loans, 1,189 of which defaulted. Therefore, the prior default rate is 19.95%. In the Australian (Taiwan) dataset, there are 690 (30,000) instances out of which 307 (6636) defaulted, leading to a prior default rate of 44.49% (22.12%). In the Housing dataset, there are 12 explanatory variables, two of which are nominal. The Australian dataset includes 6 numerical and 8 nominal predictors. For the Taiwan dataset, there are 23 predictors, nine of which are nominal. Tables A.2 and A.3 of the online appendix display the list of predictive variables for the Housing and Taiwan datasets, respectively. We do not provide this information for the Australian dataset, as all attribute names and values have been changed to meaningless symbols to maintain the confidentiality of the data.

We rely on the same $(N \times 2)$ comparison setup as for the benchmark Kaggle dataset, with $N = 5$. Table 4 displays the values of the five statistics retained for the comparison of the alternative models. For the Australian dataset, the two best performing models are PLTR and the random forest method, with similar values for all five statistics.[24] This finding once again confirms the relevance of our approach in terms of statistical performance. The same picture is observed for the Taiwan dataset with the PLTR model appearing as efficient as the random forest method.

Lastly, for the Housing dataset, the random forest method and PLTR are once again the best performing models. However, in contrast to the results obtained for the other datasets, the random forest model outperforms our method. Table 5 displays the interpretability performance for these three additional datasets. Using the same arguments as above, the average number of active variables (univariate and bivariate threshold effects) in our penalised logistic regression is equal to 47.6, while the random forest method relies on an average of 343.8×110.5 binary variables for prediction.[25] Moreover, the results of PLTR are close to those of linear logistic regression for both criteria, indicating that the PLTR model remains interpretable despite including non-linear effects. Other results, available upon request, show that by relaxing the constraint of parsimony via the inclusion of trivariate and quadrivariate threshold effects, the performance of our penalised logistic regression increases and reaches that of the random forest model. This suggests that complex non-linear relationships that go beyond univariate and bivariate threshold effects are present in this dataset. In view of this result, it is important to emphasise that our method offers a highly flexible framework to credit risk managers, as they can tune their model according to the desired level of parsimony. The predictive performance can be significantly improved but at the cost of less interpretable results.

Additional robustness results consists in out-of-sample forecasting performance comparison tests of the three main competing models, i.e. linear logistic regression, random forest, and PLTR. We rely on Diebold-Mariano (Diebold & Mariano, 1995) and AUC tests (Candelon, Dumitrescu, & Hurlin, 2012) to perform pairwise comparisons and on the Model Confidence Set (Hansen, Lunde, & Nason, 2011) to identify the bucket of models that are superior to the remaining ones and which exhibit similar performance. They are all well known model comparison approaches, the second being specific to the case with binary dependent variables. The pairwise tests are two-sided, the null hypothesis corresponds to equal performance and its rejection indicates that the model with smaller average loss is better. At the same time, the Model Confidence Set identifies the subset of models that exhibit similar forecasting abilities and outperform the remaining approaches.

Tables C.1 and C.2 in Section C of the online appendix display these results for the four datasets under analysis. They take the form of percentage of rejection of each null hypothesis in the 5×2 cross-validation test samples and the outperforming model under

[23] StatLog is an international project that aims to compare the performances of machine learning, statistical, and NN algorithms on datasets from real-world industrial areas, including medicine, finance, image analysis, and engineering design.

[24] For the non-linear logistic regression results, we find that all fitted probabilities are higher than 0.6. Therefore, as we compute the PGI within $(0, 0.4)$, this statistic cannot be computed. Unlike in practice, this bad performance can also be observed through the high value of the BS statistic compared to those of the other methods.

[25] In this dataset, we identify on average of 110.5 trees in the forest, with an average number of terminal nodes equal to 343.8 for each tree. Furthermore, at most, 18.8 predicates are used on average in the rules of the random forest method against 2 at most for the PLTR model. Hence, PLTR is once again better from the interpretability point of view.

Table 4
Statistical performance indicators: robustness check.

Methods	AUC	PGI	PCC	KS	BS
Australian dataset					
Linear Logistic Regression	0.8998	0.5664	0.8374	0.7135	0.1186
Non-Linear Logistic Regression	0.6090		0.6067	0.2266	0.3921
Non-Linear Logistic Regression + ALasso	0.8866	0.5092	0.8214	0.6816	0.1333
Random Forest	0.9344	0.6246	0.8603	0.7523	0.0999
PLTR	0.9299	0.6370	0.8606	0.7425	0.1029
Support Vector Machine	0.9210	0.5557	0.8445	0.7391	0.1122
Neural Network	0.9141	0.5799	0.8539	0.7366	0.1102
Taiwan dataset					
Linear Logistic Regression	0.6310	0.2099	0.7586	0.2506	0.2344
Non-Linear Logistic Regression	0.5963	0.0984	0.7035	0.1927	0.2965
Non-Linear Logistic Regression + ALasso	0.7596	0.5029	0.7871	0.3926	0.1447
Random Forest	0.7722	0.4924	0.8102	0.4177	0.1362
PLTR	0.7780	0.5156	0.7959	0.4257	0.1352
Support Vector Machine	0.7102	0.3207	0.8195	0.3382	0.1461
Neural Network	0.7304	0.4226	0.7879	0.3885	0.1401
Housing dataset					
Linear Logistic Regression	0.7904	0.5508	0.8103	0.4450	0.1228
Non-Linear Logistic Regression	0.7965	0.5425	0.8239	0.4650	0.1199
Non-Linear Logistic Regression + ALasso	0.8113	0.5754	0.8217	0.4815	0.1125
Random Forest	0.9387	0.8157	0.9036	0.7455	0.0736
PLTR	0.9011	0.7341	0.8818	0.6694	0.0844
Support Vector Machine	0.7890	0.5514	0.8093	0.4444	0.1254
Neural Network	0.7910	0.5478	0.8132	0.4470	0.1208

Note: Non-linear logistic regression includes linear, quadratic and interaction terms. The method labelled "Non-Linear Logistic Regression + ALasso" corresponds to a penalised version of non-linear logistic regression with the adaptive lasso penalty.

Table 5
Measures of interpretability: robustness check.

Methods	Size of the decision set	Maximal number of predicates
Australian dataset		
Linear Logistic Regression	34.4	1
Random Forest	52.4 × 69.6	8
PLTR	25.4	2
Taiwan dataset		
Linear Logistic Regression	78.7	1
Random Forest	2,378.7 × 174.7	29.9
PLTR	79.9	2
Housing dataset		
Linear Logistic Regression	17	1
Random Forest	343.8 × 110.5	18.8
PLTR	47.6	2

Note: This table displays the average values of interpretability measures for linear logistic regression, the random forest method and PLTR.

the alternative hypothesis is displayed below in parentheses. Two different loss functions are used for the general tests (Diebold-Mariano and Model Confidence Set), namely the Brier Score and the opposite of the log-likelihood, in the spirit of a robustness check.

All findings are consistent with those already obtained with statistical performance indicators. Namely, the pairwise comparisons reveal that the PLTR method is superior to standard logistic regression, and its performance is far better than that of random forests in two datasets, in the other two the results being more mitigated. Additionally, the Model Confidence Set identifies most often the PLTR method as that belonging to the subset of outperforming models.

6. Economic evaluation

An important question for a credit risk manager is to what extent these out-of-sample statistical performance gains have a positive impact at a financial level for a credit company. An economic evaluation method consists of estimating the amount of regulatory capital induced by the estimated probabilities of default. A similar comparison approach was proposed by Hurlin, Leymarie, & Patin (2018) for loss-given-default (LGD) models. However, this approach requires computing other Basel risk parameters, in particular the LGD and the exposure at default (EAD), and hence needs specific information about the consumers and the terms of the loans, which is not publicly available.

An alternative approach consists of comparing the misclassification costs (see Viaene & Dedene, 2004). This cost is estimated from Type 1 and Type 2 errors weighted by their probability of occurrence. Formally, let C_{FN} be the cost associated with a Type 1 error (the cost of granting credit to a bad customer) and C_{FP} be the cost associated with a Type 2 error (e.g., the cost of rejecting a good customer). Thus, the misclassification error cost is defined as

$$MC = C_{FP}FPR + C_{FN}FNR,$$

where FPR is the false positive rate and FNR is the false negative rate. There is no consensus in the literature about how to determine C_{FN} and C_{FP}. Two alternatives have been proposed. The first method fixes these costs by calibration based on previous studies (Akkoc, 2012). For example, West (20 0 0) set C_{FN} to 5 and C_{FP} to 1.The second method evaluates misclassification costs for differ- ent values of C_{FN} to test as many scenarios as possible (Lessmann et al., 2015). Although there is no consensus on how to determine these costs, it is generally acknowledged that the cost of granting credit to a bad customer is higher than the opportunity cost of rejecting a good customer (see Baesens et al., 2003; Thomas et al., 2002; West, 2000, among others). We choose to follow the second approach to assess the performance of the competing models. We fix C_{FP} at 1 without loss of generality (Hernández-Orallo, Flach, & Ramirez, 2011) and consider values of C_{FN} between 2 and 50. Once these misclassification costs are computed, we set the linear logistic regression as the reference, and we compute the financial gain or cost reduction associated with an alternative scoring model relative to this reference.[26]

Figures C.3-C.6 in the online appendix display the average cost reduction or financial gains over the test samples for the four datasets considered above. First, all methods deliver positive cost reductions, except in three cases. This means that financial institutions relying on each of these methods rather than on the benchmark linear logistic regression are expected to save an amount equivalent to the cost of rejecting (accepting) good (bad) applicants. In view of the large number of credits in bank credit portfolios, these gains could represent substantial savings for credit institutions. The fact that non-linear logistic regression leads to an increase in costs compared to the linear logistic regression comes from the relatively high number of variables in the two datasets (14 and 23 in the Australian and Taiwan datasets, respectively). This leads to a proliferation of predictors (squares of the variables, cross-products of the variables) and therefore to overfitting. The penalised version of the non-linear logistic regression succeeds in dealing with this issue, which materialises in positive values of cost reductions in all cases except for the Australian dataset. The NN and SVM both reduce the misclassification costs compared to the logistic regression. This result is once again consistent with the results of the literature.

Second, across all datasets, the PLTR method is among the most efficient in terms of cost reduction. For the Kaggle dataset, the cost reduction relative to the linear logistic regression is equal to 18. 06% on average. This result also holds in the Taiwan dataset, with an average cost reduction of 22. 29%. Note that the random forest method leads to lower cost reduction for these two datasets, with an average cost reduction of 13. 09% (11. 51%) for the Kaggle (Taiwan) dataset. This means that although the random forest method has high global predictive accuracy, as given by the proportion of correct classification (see Tables 1 and 4), it fails to some extent to detect bad customers, which leads to a relative increase in costs due to more false negatives. For the other two datasets (Australian and Housing), the random forest method performs well. With the Australian dataset, the average cost reduction of the random forest (PLTR) method is equal to 22. 71% (14. 89%). For the Housing dataset, the average values are equal to 44. 56% and 38. 69% for the random forest method and PLTR, respectively.[27]

To conclude, all results show that the PLTR model may generate important cost reductions compared to the standard logistic regression model generally used by the credit risk industry while preserving its intrinsic interpretability.

7. Conclusion

Despite the development and dissemination of many efficient machine learning classification algorithms, the benchmark scoring model in the credit industry remains logistic regression. This current state is caused mainly by the stability and robustness of the logistic regression model and also its intrinsic interpretability. Many academic papers advocate the use of more sophisticated ensemble methods, such as the random forest method. These black-box models are not interpretable, but many agnostic methods can be used to make their forecasting rules interpretable ex post for the various stakeholders (risk modellers, model checkers, clients, management, regulators, etc.). Nevertheless, these alternative models are still generally considered as challenger models and rarely used in the credit granting process or for regulatory purposes.

Recognising that traditional logistic regression underperforms random forest due to its pitfalls in modelling non-linear (threshold and interaction) effects, this article introduces penalised logistic tree regression (PLTR) with predictive variables given by easy-to-interpret endogenous univariate and bivariate threshold effects. These effects are quantified by dummy variables associated with leaf nodes of short-depth decision trees built with singletons and couples of the original predictive variables.

Monte Carlo simulations and an empirical application based on four real-life credit scoring datasets show that PLTR has good predictive power while remaining easily interpretable. More precisely, using several metrics and diagnostic tests to evaluate the accuracy and the interpretability of credit models, we show that it performs better in out-of-sample than traditional linear and non-linear logistic regression, while being competitive relative to the random forest method. We also evaluate the economic benefit of using our PLTR method through misclassification costs and expected maximum profit analysis. We find that beyond parsimony, the PLTR method leads to a significant reduction in misclassification costs.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ejor.2021.06.053

References

ACPR (2020). *Governance of artificial intelligence in finance*. Discussion papers publication, November, 2020

Akkoc, S. (2012). An empirical comparison of conventional techniques, neural networks and the three stage hybrid adaptive neuro fuzzy inference system (ANFIS) model for credit scoring analysis: The case of turkish credit card data. *European Journal of Operational Research, 222*(1), 168–178.

Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance, 23*(4), 589–609.

Baesens, B., Gestel, T. V., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society, 54*, 627–635.

Bauweraerts, J. (2016). Predicting bankruptcy in private firms: Towards a stepwise regression procedure. *International Journal of Financial Research, 7*(2), 147–153.

Bracke, P., Datta, A., Jung, C., & Sen, S. (2019). *Machine learning explainability in finance: An application to default risk analysis*. Bank of England. Staff Working Paper No. 816

Breiman, L. (1996). Bagging predictors. *Machine Learning, 26*, 123–140.

Breiman, L. (2001). Random forest. *Machine Learning, 45*, 5–32.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review, 78*(1).

Bussman, N., Giudici, P., Marinelli, D., & Papenbrock, J. (2019). Explainable machine learning in credit risk management. Working paper SSRN, available at doi:10.2139/ssrn.3506274.

Candelon, B., Dumitrescu, E.-I., & Hurlin, C. (2012). How to evaluate an early-warning system: Toward a unified statistical framework for assessing financial crises forecasting methods. *IMF Economic Review, 60*(1), 75–113.

[26] The misclassification costs are computed from test samples.

[27] We also consider a second measure of performance, namely, the expected maximum profit (EMP) introduced by Verbraken, Bravo, Weber, & Baesens (2014), to compare the models from an economic viewpoint. The results of this robustness exercise are displayed in Section B of the online appendix.

Cardell, N. S., & Steinberg, D. (1998). The hybrid-CART logit model in classification and data mining. *Working paper, Salford-System*.

Chan, K. Y., & Loh, W. Y. (2004). Lotus: An algorithm for building accurate and comprehensible logistic regression trees. *Journal of Computational and Graphical Statistics, 13*(4), 826–852.

Coffman, J. (1986). The proper role of tree analysis in forecasting the risk behavior of borrowers. *Management Decision Systems, Atlanta, MDS Reports, 3*(4), 7.

De Caigny, A., Coussement, K., & De Bock, K. W. (2018). A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. *European Journal of Operational Research, 269*(2), 760–772.

Deng, H. (2019). Interpreting tree ensemble with intrees. *International Journal of Data Science and Analytics, 7*(4), 277–287.

Desai, V. S., Crook, J. N., & Overstreet Jr, G. A. (1996). A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research, 95*(1), 24–37.

Diebold, F., & Mariano, R. (1995). Comparing predictive accuracy. *Journal of Business and Economic Statistics, 13*, 253–263.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation, 10*(7), 1895–1923.

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

EBA (2020). *Report on big data and advanced analytics*. European Banking Authority, January, 2020.

EC (2020). *White paper on artificial intelligence: A European approach to excellence and trust*. European Commission, February, 2020.

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood and its Oracle properties. *Journal of the American Statistical Association, 96*(456), 1348–1360.

Finlay, S. (2011). Multiple classifier architectures and their application to credit risk assessment. *European Journal of Operational Research, 210*(2), 368–378.

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software, 33*(1), 1–22.

Frost, J., Gambacorta, L., Huang, Y., Shin, H. S., & Zbinden, P. (2019). Bigtech and the changing structure of financial intermediation. *Economic Policy*.

Grennepois, N., Alvirescu, M. A., & Bombail, M. (2018). *Using random forest for credit risk models*. Deloitte Risk Advisory, September, 2018.

Grennepois, N., & Robin, E. (2019). *Explain artificial intelligence for credit risk management*. Deloitte Risk Advisory, July, 2019.

Hand, D. J. (2005). Good practice in retail credit scorecard assessment. *Journal of the Operational Research Society, 56*(9), 1109–1117.

Hansen, P. R., Lunde, A., & Nason, J. M. (2011). The model confidence set. *Econometrica, 79*(2), 453–497.

Henley, W., & Hand, D. (1996). A k-nearest-neighbour classifier for assessing consumer credit risk. *The Statistician, 45*(1), 77–95.

Henley, W. E., & Hand, D. J. (1997). Construction of a k-nearest neighbour credit-scoring system. *IMA Journal of Mathematics Applied in Business and Industry, 8*, 305–321.

Hernández-Orallo, J., Flach, P. A., & Ramirez, C. F. (2011). Brier curves: a new cost-based visualisation of classifier performance.. In *Icml* (pp. 585–592).

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55–67.

Hurlin, C., Leymarie, J., & Patin, A. (2018). Loss functions for loss given default model comparison. *European Journal of Operational Research, 268*(1), 348–360.

Hurlin, C., & Pérignon, C. (2019). Machine learning et nouvelles sources de données pour le scoring de crédit. *Revue d'économie financière*, (3), 21–50.

Landwehr, N., Hall, M., & Frank, E. (2005). Logistic model trees. *Machine Learning, 59*, 161–205.

Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research, 247*, 124–136.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 4765–4774.

Makowski, P. (1985). Credit scoring branches out. *Credit World, 75*(1), 30–37.

Martens, D., Baesens, B., & Van Gestel, T. (2008). Decompositional rule extraction from support vector machines by active learning. *IEEE Transactions on Knowledge and Data Engineering, 21*(2), 178–191.

Matignon, R. (2007). *Data mining using SAS enterprise miner*.

McIlhagga, W. H. (2016). Penalized: A Matlab toolbox for fitting generalized linear models with penalties.

Molnar, C. (2019). Interpretable machine learning. In *A guide for making black box models explainable*. https://christophm.github.io/interpretable-ml-book/

Óskarsdóttir, M., Bravo, C., Sarraute, C., Vanthienen, J., & Baesens, B. (2019). The value of big data for credit scoring: Enhancing financial inclusion using mobile phone data and social network analytics. *Applied Soft Computing, 74*, 26–39.

Paleologo, G., Elisseeff, A., & Antonini, G. (2010). Subagging for credit scoring models. *European Journal of Operational Research, 201*(2), 490–499.

Park, M. Y., & Hastie, T. (2007). L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society, 69*(4), 659–677.

Pundir, S., & Seshadri, R. (2012). A novel concept of partial Lorenz curve and partial Gini index. *International Journal of Engineering, Science and Innovative Technology, 1*, 296–301.

Setiono, R., Baesens, B., & Mues, C. (2008). Recursive neural network rule extraction for data with mixed attributes. *IEEE Transactions on Neural Networks, 19*(2), 299–307.

Shewchuk, J. R., et al. (1994). *An introduction to the conjugate gradient method without the agonizing pain*. Carnegie-Mellon University. Department of Computer Science.

Srinivasan, V., & Kim, Y. H. (1987). Credit granting: A comparative analysis of classification procedures. *The Journal of Finance, 42*(3), 665–681.

Tam, K. Y., & Kiang, M. Y. (1992). Managerial applications of neural networks: The case of bank failure predictions. *Management Science, 38*(7), 926–947.

Thomas, L., Crook, J., & Edelman, D. (2017). *Credit scoring and its applications*. SIAM.

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.

Thomas, L. C., Edelman, D. B., & Crook, J. N. (2002). *Credit scoring and its application*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, 58*(1), 267–288.

Vapnik, V. N., & Chervonenkis, A. Y. (1971). On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications, 2*(16), 264–280.

Verbraken, T., Bravo, C., Weber, R., & Baesens, B. (2014). Development and application of consumer credit scoring models using profit-based classification measures. *European Journal of Operational Research, 238*(2), 505–513.

Viaene, S., & Dedene, G. (2004). Cost-sensitive learning and decision making revisited. *European Journal of Operational Research, 166*, 212–220.

Wang, W. (2012). How the small and medium-sized enterprises' owners' credit features affect the enterprises' credit default behavior? *E3 Journal of Business Management and Economics, 3*(2), 90–95.

West, D. (2000). Neural network credit scoring models. *Computers & Operations Research, 27*(11–12), 1131–1152.

Yobas, M. B., Crook, J. N., & Ross, P. (2000). Credit scoring using neural and evolutionary techniques. *IMA Journal of Mathematics Applied in Business and Industry, 11*, 111–125.

Zhang, H. H., & Lu, W. (2007). Adaptive lasso for Cox's proportional hazards model. *Biometrika, 94*(3), 691–703.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association, 101*(476), 1418–1429.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, 67*(2), 301–320.