GAM(L)A: An econometric models for interpretable Machine Learning

Emmanuel Flachaire*, Gilles Hacheme†, Sullivan Hué‡ and Sébastien Laurent§

December 29, 2021

Abstract

Despite their high predictive performance, machine learning (ML) algorithms are often considered as black boxes or uninterpretable models which has raised concerns from practitioners and regulators. We propose in this paper a class of partial linear models that are inherently interpretable that also competes with sophisticated ML algorithms in terms of predictive performance. Specifically, this article introduces GAM-lasso (GAMLA) and GAM-autometrics (GAMA) models, which combine parametric and non-parametric functions to accurately capture linearities and non-linearities prevailing between dependent and explanatory variables, into a variable selection procedure to control for overfitting issues. We illustrate the predictive performance and interpretability of GAMLA and GAMA from a regression and a classification problem. The results show that GAMLA and GAMA outperform parametric models and parametric models augmented by quadratic, cubic and interaction effects. Moreover, the results also suggest that the performance of our new models is not significantly different from that of benchmark ML algorithms and is even better in some cases.

*Aix-Marseille University (Aix-Marseille School of Economics), CNRS & EHESS

†Aix-Marseille University (Aix-Marseille School of Economics), CNRS & EHESS,

‡Aix-Marseille University (Aix-Marseille School of Economics), CNRS & EHESS,

§Aix-Marseille University (Aix-Marseille School of Economics), CNRS & EHESS, Aix-Marseille Graduate School of Management – IAE, France. The authors acknowledge research support by the French National Research Agency Grants ANR-17-EURE-0020 and ANR-21-CE26-0007-01 (project MLEforRisk) and by the Excellence Initiative of Aix-Marseille University - A*MIDEX.

Contents

1 Introduction

In recent years, machine learning (ML) algorithms have received considerable attention in the literature and overshadowed traditional econometric models in most applications. Although econometrics and ML have developed in parallel, both approaches share the common goal of building predictive models. For that purpose, econometrics relies on probabilistic models describing economic phenomena, whereas ML builds upon smart algorithms learning on their own. However, ML algorithms have recently been shown to be more effective than traditional econometric approaches for modelling complex relationships (Varian, 2014; Lessmann et al., 2015; Charpentier et al., 2018; Gunnarsson et al., 2021). Indeed, unlike traditional econometric models, these algorithms are able to capture many complex non-linear relationships through non-parametric approaches, resulting in higher predictive performance. The dominance of ML models in terms of predictive performance, in addition to several other advantages, has led these techniques to be used in several industries. For example, banks and fintech firms are currently considering ML algorithms as challenger models (ACPR, 2020) in the context of credit scoring, and in some cases ML models are even used for credit production (Hurlin and Pérignon, 2019).

However, ML algorithms raise a very important issue for the industry due to their lack of interpretability. Indeed, most of these algorithms are generally considered to be "black-boxes", i.e., the opacity of ML techniques leads users to predictions and decision processes that cannot be easily interpreted. The lack of interpretability is currently one of the main limitations of ML algorithms and raises concerns in many applications such as medicine, law, military or finance. ML algorithms need to be interpretable to justify predictions made by the models. For example, in the financial industry, executives need to be able to understand the model to justify their decisions, and regulators require interpretability to ensure that the algorithm used is fair.[1] Furthermore, the lack of interpretability of ML algorithms is currently one of the major concerns of financial regulators regarding the governance of artificial intelligence approaches in the financial industry (Bracke et al., 2019; ACPR, 2020; EBA, 2020; EC, 2020).

To address this issue, the literature has recently focused on interpretable ML methods.[2] Specifically, many model-agnostic methods have been proposed to interpret the ex post predictions of black-box models. For example, we can cite here the partial dependence plot (Friedman, 2001), accumulated local effect (Apley and Zhu, 2020), local interpretable model-agnostic explanations (Ribeiro et al.,

[1] See Barocas et al. (2018) for more details on the fairness of ML techniques in a general context and Hurlin et al. (2021) and Kozodoi et al. (2021) in the context of credit scoring.

[2] Interpretable ML methods seek to explain the behaviour and predictions of ML algorithms. See Molnar et al. (2020) for more details on interpretable ML.

2016) or SHAP (Lundberg and Lee, 2017).[3] However, interpretations obtained from these methods can be inaccurate representations of the original relationships and potentially mislead users to accept incorrect recommendations, which can be harmful in a high-stakes decision-making context (Rudin, 2019).

Within this context, we propose in this paper a class of models that are inherently interpretable. Instead of explaining the predictions of black boxes, we design models that are fundamentally interpretable and compete with ML algorithms in terms of predictive performance. Denoted as GAM-lasso (GAMLA) and GAM-autometrics (GAMA), these models combine the predictive performance of ML approaches with the inherent interpretability of econometric models. Formally, this class of models is based on a generalized additive model (GAM) augmented by linear interaction effects. However, due to the possibly large number of interaction effects, we perform variable selection on interactions to avoid overfitting issues. For that purpose, we rely on the lasso (Tibshirani, 1996) and autometrics (Doornik et al., 2009) approaches. Finally, as the models involve linear (interaction effects) and non-linear (smooth functions of GAM) terms, the variable selection is performed based on the double residual approach of Robinson (1988).

This class of models has several advantages. First, these models are fundamentally interpretable, unlike ML algorithms. Indeed, GAMLA and GAMA inherit the simplicity of interpretation of traditional econometric models. Specifically, while smooth functions allow a simple interpretation of the estimated relationships prevailing between dependent and predictive variables, interaction effects can be interpreted as in a simple linear model because they are introduced linearly. Moreover, the importance of predictive variables can easily be measured from the marginal effects, as in standard econometric models. GAMLA and GAMA thus allow a simple interpretation of prediction and decision processes, unlike ML algorithms. This class of models is also consistent with the recent literature promoting inherently interpretable models instead of interpretable ML methods (Rudin, 2019; Rudin and Radin, 2019; Rudin et al., 2021).

Second, GAMLA and GAMA are able to compete with ML algorithms in terms of predictive performance. Indeed, the combination of smooth functions and interaction effects allows them to capture different types of non-linear effects, i.e., non-linear effects in covariates and their interactions, similarly to ML algorithms. Specifically, GAMLA and GAMA can even outperform ML algorithms. Indeed, the recent success of ML algorithms has made the literature forget that parametric models can outperform non-parametric models in terms of predictive power if they are well specified. Therefore, one objective of this approach is also to reconcile the literature with parametric models by showing that a flexible and accurate parametrization of parametric models can lead to high predictive performance.

Third, our approach can also be considered a systematisation of a common

[3]See Molnar (2019) for an overview of interpretable ML methods.

method used by many practitioners to improve the predictive performance of linear models. Indeed, practitioners usually include ad hoc parametric functions to capture non-linearities, such as quadratic, cubic or interaction effects (Hurlin and Pérignon, 2019). GAMLA and GAMA thus share some similarities with the penalised logistic tree regression (PLTR) model of Dumitrescu et al. (2021), developed in the context of credit scoring. Indeed, both models propose to improve the predictive performance of linear models by automatically capturing non-linear effects. However, while the PLTR is based on univariate and bivariate threshold effects obtained from short-depth decision trees, GAMLA and GAMA rely on smooth functions and interaction effects.

We provide a set of Monte Carlo experiments to assess the predictive performance and variable selection consistency of our new class of models. The results suggest that GAMLA and GAMA accurately capture non-linear relationships, unlike the approaches commonly employed by practitioners, and are competitive with benchmark ML algorithms. Moreover, we show that the double residual approach of Robinson (1988) leads GAMLA and GAMA to consistent selection of variables. The results also suggest the use of GAMA because it allows for the selection of a large number of relevant variables while controlling for the number of irrelevant variables, unlike GAMLA.

Finally, we illustrate the practical usefulness of GAMLA and GAMA using data on regression and classification problems. To that end, we measure the predictive performance of our models using popular measures of performance as well as inference procedures and compare them to the benchmark models in the econometrics and ML literatures. We show that GAMLA and GAMA achieve higher predictive performance than linear models, even when augmented by parametric functions used by practitioners. Moreover, the results suggest that our models compete with sophisticated ML algorithms in terms of predictive performance and outperform them in some cases. We also illustrate the interpretability of our new class of models. For that purpose, we assess the parsimony of GAMLA and GAMA and show the simple interpretation of estimated relationships and decision rules through graphical representations of smooth functions and marginal effects. The results suggest that GAMLA and GAMA remain interpretable despite capturing complex non-linear relationships, unlike ML algorithms.

The remainder of the article is structured as follows. In Section 2, we present the main advantages of ML algorithms as well as the main limitations of traditional linear models and introduce our new class of interpretable models. Section 3 is devoted to Monte Carlo experiments. In Section 4, two empirical applications are proposed to illustrate the potential of GAMLA and GAMA. Finally, we conclude the article in Section 5.

2 Competing with Black Boxes: An Interpretable Parametric Model

In this section, we present a class of partial linear models that is able to compete with sophisticated ML algorithms in terms of predictive performance while remaining interpretable. The first part of the section describes why ML algorithms achieve high predictive performance and provides a brief presentation of current benchmark ML algorithms. The second part presents the pitfalls of parametric models in that respect. Finally, the last part of the section is devoted to the presentation of our new class of partial linear models.

2.1 Machine Learning, Non-linear Effects and Interactions

Consider a regression problem involving a dependent variable $y \in \mathbb{R}$ and a p-dimensional vector of predictive variable $X = (X_1, \ldots, X_p) \in \mathbb{R}^p$. Recent years have witnessed a paradigm shift in terms of the models used to make predictions. Indeed, ML algorithms have progressively replaced parametric models in many applications where the main objective is to build accurate predictions. In general, ML algorithms can be defined such as

$$y = f(X) + \epsilon, \tag{1}$$

where $f(.)$ is a non-parametric function and ϵ an i.i.d. error term with zero mean and variance σ_ϵ^2. The main reason for the increased use of ML models is that these algorithms lead to very high predictive performance and currently outperform parametric models traditionally used by practitioners. The hegemony of ML algorithms over parametric models has been highlighted in several contexts, such as in the credit scoring literature (Paleologo et al., 2010; Finlay, 2011; Lessmann et al., 2015).[4] The high performance of ML algorithms derives from the fact that the non-parametric functions $f(.)$ used by these approaches are able to accurately capture complex non-linear effects of covariates and interaction effects. Specifically, instead of specifying a certain relationship between the dependent and explanatory variables, these non-parametric functions $f(.)$ rely almost exclusively on data to detect non-linearities and interaction effects prevailing between y and X.

Among the many algorithms proposed in the ML literature, ensemble methods such as the random forest (Breiman, 2001) and gradient boosting (Friedman, 2001) have been shown to lead to very accurate predictions and even become benchmark models in terms of predictive performance (Lessmann et al., 2015; Grennepois

[4]Credit scoring is one of the first fields to which ML algorithms were applied in economics. See, for instance, Makowski (1985), Henley and Hand (1996), Desai et al. (1996), and Baesens et al. (2003).

et al., 2018; Gunnarsson et al., 2021). Random forest and gradient boosting are particular applications of bagging and boosting procedures, which are based on the aggregation of weak learners, such as decision trees.[5] The decision tree algorithm is based on a recursive partition of the initial data into smaller homogeneous subsets, in the sense of the dependent variable. Specifically, decision trees recursively split the covariate space into two homogeneous partitions, called nodes, until obtaining nodes that are as homogeneous as possible, which are called terminal nodes or leaves.[6] To do so, the algorithm chooses for each partition the most discriminant explanatory variable, i.e., the variable partitioning the original node into the two most homogeneous nodes possible. Formally, a decision tree is defined as

$$f_{Dt}(X) = \sum_{m=1}^{M} c_m \mathbb{I}(X \in R_m), \tag{2}$$

where M is the total number of leaves of the tree, R_m is a leaf of the tree and c_m corresponds to the average of the observations' dependent variable in R_m. Despite capturing threshold effects through multiple splits, the predictive performances of decision trees is lacklustre and barely better than random guessing due to their high variance. To solve this issue and achieve higher performance, ensemble methods such as random forest and gradient boosting aggregate several decision trees.[7] On the one hand, the random forest algorithm is based on the combination of several decision trees fitted on copies of the original data obtained from a bootstrap procedure.[8] Denoting by B the total number of trees in the forest, which also corresponds to the number of bootstrap samples, a random forest is defined as

$$f_{Rf}(X) = \frac{1}{B} \sum_{b=1}^{B} f_{Dt}^{b}(X), \tag{3}$$

where $f_{Dt}^{b}(X)$ is a decision tree fitted on the b^{th} bootstrap sample. Predictions of the random forest thus simply correspond to averages of individual decision tree predictions. On the other hand, the gradient boosting algorithm builds decision trees sequentially, each one learning information from previous decision trees. Unlike the random forest, gradient boosting does not involve a bootstrap sampling strategy, but instead, each tree is built upon the errors of the previous decision

[5] Weak learners are models that perform slightly better than random predictions.

[6] The binary partition corresponds to the CART algorithm (Breiman et al., 1984), which is the most popular decision tree algorithm.

[7] By doing so, these algorithms become strong learners, i.e., models that perform substantially better than random predictions.

[8] In the random forest algorithm, both observations and explanatory variables are randomly selected, whereas in the general bagging procedure, only the observations are.

tree. Formally, gradient boosting is defined as

$$f_{Gb}(X) = \sum_{b=1}^{B} \lambda_b f_{Dt}^b(X), \tag{4}$$

where B is still the total number of trees grown, which also corresponds to the number of iterations of the algorithm, and λ_b represents the weight attributed to the b^{th} tree. From the aggregation of several decision trees, random forest and gradient boosting thus make it possible to capture many complex non-linearities and interaction effects, which explain their high predictive performances.

2.2 Pitfalls of Parametric Models

Historically, the simplest approach used by practitioners to predict y is to rely on the following parametric model

$$y = X\beta + \epsilon, \tag{5}$$

where β is the set of parameters to estimate. This simple model assumes a linear relationship between the dependent variable and the predictive variables. However, in practice, the true relationship prevailing between y and X may not be linear but more complex, involving non-linearities and interaction effects. For this reason, the performance of this simple parametric model is lacklustre compared to those of the random forest or gradient boosting algorithms in the presence of complex relationships. To avoid this pitfall of the simple parametric model, a common approach used by many practitioners is to augment Eq.(5) with parametric functions of X, such as

$$y = X^*\beta + \epsilon, \tag{6}$$

where X^* is a K-dimensional vector of predictive variables obtained from transformations of X. The objective of X^* is to capture potential non-linear and interaction effects through parametric transformations of X. A typical example of parametric functions frequently used by practitioners to capture non-linearities and interaction effects are the quadratic and cubic functions as well as interactions between covariates,[9] such as

$$X^* = \left(X_1, \ldots, X_p, X_1^2, \ldots, X_q^2, X_1^3, \ldots, X_q^3, X_1X_2, \ldots, X_{q-1}X_q\right), \tag{7}$$

where $q \leqslant p$.[10]

[9] It is also possible to include principal components of explanatory variables to capture non-linear effects. See, for example, Castle and Hendry (2010) and Castle et al. (2013).

[10] This condition allows the researcher to exclude meaningless transformations of X. For example, it is unnecessary to consider quadratic or cubic functions of a dummy variable.

However, this approach is subject to overfitting issues. Indeed, the model can involve a very large number of parameters to estimate because the number of predictors depends on the number of original explanatory variables p. For example, Eq.(7) involves $K = 75$ predictors for $p = q = 10$. Consequently, performing traditional estimation on this high-dimensional model obviously leads to overfitting. To solve this issue, several approaches have been proposed in the literature to reduce the number of parameters to estimate.

An early approach developed in the literature is lasso. Proposed in the seminal paper of Tibshirani (1996), lasso is a penalized regression that precedes both estimation and variable selection. To do so, the approach is based on a penalty term that regularises coefficients and performs variable selection as the penalty function is convex and non-differentiable at the origin. Considering the linear model associated with Eq.(6), lasso solves the following penalized regression problem

$$\hat{\beta} = \arg\min_{\beta} \left[(y - X^*\beta)^T (y - X^*\beta) + \lambda \sum_{k=1}^{K} |\beta_k| \right]. \tag{8}$$

The parameter λ is a tuning parameter controlling for the degree of regularisation and is generally selected via k-fold cross-validation or information criteria. For cross-validation, two approaches can be considered to select the optimal value of λ: the value minimizing the prediction error of the model, $\hat{\lambda}^{min}$, or the value associated with the most parsimonious model within a 1-standard-error interval, $\hat{\lambda}^{1se}$.[11]

An alternative approach for variable selection is autometrics (Doornik et al., 2009). This algorithm performs automatic selection model based on the "Hendry" general-to-specific model selection (Hendry, 2000). Specifically, this approach starts from a generalized unrestricted model (GUM) that includes every potential relationship existing between the dependent variable and predictors, i.e., dynamic effects, breaks, trends, outliers and non-linearities. To reduce the dimension of the GUM, the algorithm performs a battery of tests to eliminate insignificant variables and find a congruent parsimonious model. Rather than testing for each possible sub-model from the GUM, autometrics performs a tree search that reduces the computation time and allows the approach to be feasible even in a high-dimensional context. Finally, the algorithm selects the sub-model encompassing the GUM in the representation of the relationship of concern and passing a battery of diagnostic tests. The diagnostic tests are the error correlation test (Godfrey, 1978), the ARCH test (Engle, 1982), the normality test (Doornik and Hansen, 2008), the heteroscedasticity test (White, 1980) and the RESET test (Ramsey, 1969). One interesting property of autometrics is that the user can choose the target size α, which corresponds to the percentage of irrelevant variables surviving the reduction

[11] This rule of thumb was proposed by Breiman et al. (2017).

procedure. This parameter can thus be considered a tuning parameter that solely depends on the user's leniency regarding irrelevant variables. For example, a liberal choice could be to fix the target size $\alpha = 0.05$, whereas a more conservative user might prefer lower values, such as $\alpha = 0.01$.

2.3 A Partial Linear Approach: GAM-lasso and GAM-autometrics

Although variable selection methods can address overfitting issues, large differences can still be observed between performances of ML algorithms and augmented parametric models. Indeed, the parametric transformations considered in Eq.(6) assume specific non-linear relationships prevailing between y and X, which may not be the true relations. Therefore, if differences are observed between the two types of models, this implies that the parametric functions failed to capture the true non-linear effects, resulting in lower predictive performance. Moreover, if non-linearities are not accurately captured by parametric transformations, it may also disrupt the estimation of interaction effects, leading to an inconsistent selection of these interactions.

For that reason, we propose in this paper the following class of partial linear models:

$$y = Z\gamma + \sum_{j=1}^{p} g_j\left(X_j\right) + \epsilon, \tag{9}$$

where $Z = (X_1X_2, \ldots, X_{q-1}X_q)$ is an S-dimensional vector of interactions of covariate couples with $S = (q \times (q-1))/2$, γ is the associated set of parameters, and $g_j(.)$ is a non-parametric function. This class of models allows us to accurately capture both interaction effects, introduced linearly, and non-linearities of covariates, from non-parametric functions. Therefore, models following the representation of Eq.(9) can accurately capture many complex relationships.

The representation of Eq.(9) also has the advantage of keeping the model interpretable, unlike ML algorithms. Indeed, interaction effects can be easily interpreted because they are introduced linearly in the model, whereas non-parametric functions allow one to identify non-linearities of covariates. Moreover, the linearity assumption on the interaction effects implies that marginal effects of Eq.(9) can be computed as

$$\frac{\partial y}{\partial X_j} = c + g_j'\left(X_j\right), \tag{10}$$

where $c = X_{(-j)}\gamma_j$, $X_{(-j)}$ is the $(p-1)$-dimensional vector of covariates excluding X_j, γ_j is the set of coefficients associated with the $p-1$ pairs of interactions involving X_j, and $g_j'(X_j)$ is the partial derivative of $g_j(X_j)$. This assumption simplifies the interpretation of the model because marginal effects correspond to the marginal effects of covariates taken individually, eventually augmented by a

constant corresponding to the sum of the interactions' marginal effects. One could argue about the linear introduction of interaction effects in the model and relax this assumption by considering a non-linear representation of interactions. This idea has been investigated by Chouldechova and Hastie (2015). This approach allows the researcher to potentially model both covariates and interaction effects non-linearly while controlling for overfitting issues through variable selection. Although interesting in terms of predictive performance, this approach damages the interpretability of the model. The interpretation of interaction effects becomes complicated, but most important, the marginal effects can no longer be easily computed as in Eq.(10). Indeed, the marginal effects would not correspond to standard marginal effects of covariates augmented by a constant but to a much more complex formula depending on the non-linear relationships identified for interaction effects. The linearity assumption on interaction effects thus represents the price to pay to keep the model interpretable.

To estimate non-linearities of covariates, we rely on the non-parametric functions of the generalized additive model (GAM). Introduced by Hastie and Tibshirani (2017), the GAM allows one to relax the assumption of a linear relationship between X and y and automatically captures non-linear effects through smooth functions such as

$$g_j\left(X_j\right) = \sum_{l=1}^{d} \theta_{j,l} b_{j,l}\left(X_j\right), \tag{11}$$

where $b_{j,l}\left(.\right)$ is a basis function and $\theta_j = \left(\theta_{j,1}, \ldots, \theta_{j,d}\right)$ are the associated parameters. The GAM allows for the simultaneous estimation of the parameters of parametric terms γ as well as the smooth functions $\sum_{j=1}^{p} g_j\left(X_j\right)$ using the backfitting algorithm, which is based on the following objective function:

$$\left(y - \left(Z\gamma + \sum_{j=1}^{p} g_j\left(X_j\right)\right)\right)^T \left(y - \left(Z\gamma + \sum_{j=1}^{p} g_j\left(X_j\right)\right)\right) + \psi \sum_{j=1}^{p} \int \left[g_j''\left(t\right)\right]^2 dt, \tag{12}$$

where ψ is a smoothing parameter. Similar to Eq.(8), the objective function of the GAM is penalized. However, the goal of this penalization is not to proceed to variable selection but to avoid overfitting of smooth functions. Indeed, the smoothing parameter ψ prevents non-parametric functions from becoming too wiggly, which could lead to the model having low generalizability. In practice, the smoothing parameter is generally obtained by generalized cross validation, and the larger the value of ψ is, the smoother the functions.

Similarly to Eq.(6), the large number of parameters to estimate in Eq.(9) can potentially lead to overfitting issues and thus requires proceeding to variable selection. However, as our new class of partial linear models includes both linear (interaction effects) and non-linear (smooth functions) terms, variable selection

methods cannot be applied in a standard way. Indeed, partial linear models require a specific estimation method because the presence of non-linear terms can potentially disturb the estimation of linear parameters' coefficients, leading to an inconsistent selection of linear terms. For that purpose, we propose to combine variable selection with the double residual methodology of Robinson (1988). The double residual approach follows the Frisch–Waugh–Lovell theorem (Frisch and Waugh, 1933; Lovell, 1963) and allows one to consistently estimate partial linear models. The conditional expectation of the partial linear model defined in Eq.(9) is given by

$$\mathbb{E}\left(y|X\right) = \mathbb{E}\left(Z|X\right)\gamma + \sum_{j=1}^{p} g_j\left(X_j\right). \tag{13}$$

When subtracted from Eq.(9), it leads to the following equation

$$y - \mathbb{E}\left(y|X\right) = \left(Z - \mathbb{E}\left(Z|X\right)\right)\gamma + \epsilon. \tag{14}$$

If $\mathbb{E}\left(y|X\right)$ and $\mathbb{E}\left(Z|X\right)$ are known, then Eq.(14) can be simply estimated by ordinary least squares (OLS). However, as these conditional expectations are unknown, they must be estimated by some consistent non-parametric estimators. For that purpose, we estimate conditional expectations based on the following GAM models:

$$y = \sum_{j=1}^{p} g_j\left(X_j\right) + u, \tag{15}$$

$$Z_s = \sum_{j=1}^{p} g_j\left(X_j\right) + v_s, \ \forall s, \tag{16}$$

where Z_s is the interaction of two different covariates with $s = 1, \dots S$ and u and v_s are i.i.d. error terms. Note that we can estimate Eq.(16) only if X and Z are not perfectly multicollinear, which is the case in our approach. Specifically, the following condition must hold:

$$E\left(v_s^\top v_s\right) \text{ is positive definite,}$$

where v_s is the error term of Eq.(16). Moreover, instead of estimating non-parametric functions by multivariate kernel methods as in Li and Racine (2007), we rely on univariate GAM smooth functions. We propose to apply lasso and autometrics variable selection methods to the following double residuals model

$$\hat{u} = \hat{v}\delta + \epsilon, \tag{17}$$

where $v = (v_1, \dots, v_S)$ and $\hat{u}$ and $\hat{v}$ represent residuals of Eqs.(15)-(16), respectively. The double residual approach of Robinson (1988) leads to a root-n-consistent

estimate of linear terms' parameters δ, allowing us to correctly perform variable selection on these terms.[12] Finally, the smooth functions and parameters of the selected interactions are estimated from the following GAM model:

$$y = Z^*\gamma + \sum_{j=1}^{p} g_j\left(X_j\right) + \epsilon, \tag{18}$$

where Z^* is the set of interaction effects selected from lasso or autometrics. We denote as GAM-lasso (GAMLA) the resulting model from a lasso penalization and as GAM-autometrics (GAMA) the model obtained with autometrics.[13]

3 Simulation Study

We use Monte Carlo simulations to study the potential of our new approach. Specifically, we assess the predictive performance and variable selection consistency of GAMLA and GAMA compared to those of benchmark approaches in the literature. For that purpose, the Monte Carlo simulation is performed on $n_r = 1000$ replications, with each model being trained on a training sample of $n^{in} = 1000$ observations and evaluated on $n^{out} = 1000$ out-of-sample observations, with $n = n_{in} + n_{out}$.

3.1 Simulation Design

We generate $p = 10$ predictive variables $x_{i,j}$, $j = 1, \ldots, p$, $i = 1, \ldots, n$, and consider the following data generating process (DGP)

$$y_i = \sum_{j=1}^{p}\sum_{k=j+1}^{p} \gamma_{j,k} x_{i,j} x_{i,k} + \sum_{j=1}^{p} g_j\left(x_{i,j}\right) + \epsilon_i, \tag{19}$$

where $\epsilon_i \sim \mathcal{N}\left(0,1\right)$ and $g = \left(g_1, \ldots, g_p\right)$ are some functions. The predictive variables are simulated as

$$x_{i,j} \sim \mathcal{N}\left(0,1\right) \text{ for } j = 1, \ldots, q, \tag{20}$$

$$x_{i,j} = -g_j\left(x_{i,j-q}\right)/x_{i,j-q} + u \text{ for } j = q+1, \ldots, p, \tag{21}$$

[12] Note here that we proceed to variable selection only on the linear part of Eq.(9), i.e., the interaction effects.

[13] There already exists an R package that allows one to estimate GAM models penalized by lasso (Ghosal and Kormaksson, 2019). However, this package does not apply the double residual approach.

where $q = 5$ and $u \sim \mathcal{N}(0, 0.4)$, and

$$g_j(x_{i,j}) = \begin{cases} \sin(5x_{i,j}) & \text{for } j = 1, \\ 5x_{i,j} - 5x_{i,j}\mathbb{I}(x_{i,j} > 0) & \text{for } j = 2, \\ LogN(x_{i,j}, 0.5) & \text{for } j = 3, \\ e^{x_{i,j}} & \text{for } j = 4, \\ \arctan(10x_{i,j}) & \text{for } j = 5, \\ 0 & \text{otherwise,} \end{cases} \tag{22}$$

where $LogN(.)$ is the probability density function of a log-normal distribution. Specifically, the DGP assumes that functions g are non-linear and that some explanatory variables x_i are correlated with these non-linear functions. Therefore, this DGP allows us (i) to illustrate the potential of non-parametric functions to accurately capture non-linearities compared to parametric functions and (ii) to highlight the importance of accounting for the correlation between linear and non-linear terms.

To study the consistency of variable selection, we assume that few interaction effects are relevant. Specifically, the relevant interaction effects are $x_{i,j}x_{i,j+q}$ for $j = 1, \ldots, q$. To determine their associated coefficients, we rely on non-centrality parameters rather than randomly drawing the associated coefficients. Denoting by $W = (g_1, \ldots, g_p, x_1x_2, \ldots, x_{p-1}x_p)$ the matrix of regressors, the coefficients of relevant variables are defined as

$$\gamma_{j,k} = \xi_{j,k}\sqrt{\mathbb{E}[W'W]^{-1}_{j,k}}, \tag{23}$$

where $\xi_{j,k}$ is the non-centrality parameter and $\mathbb{E}[W'W]_{j,k}$ is the term's diagonal associated with the interaction x_jx_k.[14] The non-centrality parameter $\xi_{j,k}$ allows us to calibrate the significance of the $\gamma_{j,k}$ parameters, i.e., it determines the power of student's test of significance. We consider $\xi_{j,k} = 6$ for $j = 1, \ldots, q$, and $k = j + q$, leading to a power equal to 1. Therefore, it implies that interaction effects associated with these coefficients should be identified as important by variable selection methods and thus included in the model if the variable selection is consistent. Regarding irrelevant variables, the associated coefficients are set to 0. Therefore, these variables should not be included in the model if the variable selection is consistent

In the simulation, we compare GAMLA and GAMA to several benchmark approaches in the literature.[15] We consider linear models augmented by quadratic, cubic and interaction terms, as in Eq.(7), whose variables have been selected by

[14]The value $\mathbb{E}[W'W]^{-1}$ being unknown, we estimate it by simulating n_{in} observations of the matrix W, computing $(W'W)^{-1}$ for n_r replications and taking the average over all replications.

[15]For each replication, we consider a cubic basis for non-parametric functions of GAMLA and GAMA.

lasso and autometrics. Denoted as LASSO and AM, these models allow us to illustrate the failure of parametric functions to accurately capture non-linearities, as well as the inconsistency of variable selection in the presence of both linear and non-linear terms. For both variable selection methods, we consider two rules of thumb, i.e., $\hat{\lambda}^{min}$ and $\hat{\lambda}^{1sec}$ for the lasso and $\alpha = 0.05$ and $\alpha = 0.01$ for autometrics. We include in the comparison GAMLA and GAMA whose variable selection is performed traditionally, i.e., without relying on the double residual approach of Robinson (1988). Denoted as GAMLA* and GAMA*, these models enable us to highlight the absolute need to rely on the double residual approach in the presence of both linear and non-linear terms. We also compare GAMLA and GAMA to the methodology of Chouldechova and Hastie (2015) which allows us to non-linearly model covariates and interaction effects. Denoted as GAMSEL, this approach is based on lasso to control for overfitting issues. Finally, we compare the predictive performance of GAMLA and GAMA to that of a standard OLS model and that of the current ML benchmarks, i.e., random forest and XGBoost.[16]

The model performance analysis is conducted using three criteria. First, we study the consistency of variable selection with the potency and the gauge criteria (Castle et al., 2011). Specifically, potency measures the frequency of relevant interaction variables included in the model, such as

$$\text{Potency} = \frac{1}{q} \sum_{j=1}^{q} \mathbb{I}\left(\hat{\gamma}_{j,j+q} \neq 0\right), \tag{24}$$

where $\hat{\gamma}_{j,j+q}$ is the coefficient estimated by the model associated with $x_j x_{j+q}$, whereas gauge assesses the frequency of irrelevant interactions included in the model, such as

$$\text{Gauge} = \frac{1}{S-q} \sum_{j=1}^{p} \sum_{\substack{k=j+1, \\ k \neq j+q}}^{p} \mathbb{I}\left(\hat{\gamma}_{j,k} \neq 0\right). \tag{25}$$

The optimal variable selection method is thus that including the highest percentage of relevant variables in the model, i.e., the highest potency level, while also controlling for the percentage of irrelevant variables, i.e., the gauge level. Second, we evaluate the predictive performance of models using the mean squared error, which is defined as

$$\text{MSE} = \frac{1}{n^{out}} \sum_{i=1}^{n^{out}} \left(\hat{y}_i - y_i\right)^2, \tag{26}$$

where $\hat{y}_i$ is the prediction obtained from a model for the i^{th} out-of-sample observation.

[16]The XGBoost algorithm (Chen et al., 2015) is a particular implementation of the gradient boosting algorithm presented in Section 2.1 that allows for fast computation and is currently very popular among practitioners and researchers.

3.2 Evaluation Results

Table 1: Comparison of potency, gauge and MSE under non-linear effects and correlated covariates

Model	Rule of thumb	Potency	Gauge	MSE
LASSO	$\hat{\lambda}^{min}$	0.431	0.226	1.205
	$\hat{\lambda}^{1se}$	0.020	0.010	1.223
AM	$\alpha = 0.05$	0.604	0.058	1.190
	$\alpha = 0.01$	0.433	0.022	1.181
GAMLA*	$\hat{\lambda}^{min}$	0.390	0.086	1.159
	$\hat{\lambda}^{1se}$	0.001	0.000	1.198
GAMA*	$\alpha = 0.05$	0.513	0.050	1.152
	$\alpha = 0.01$	0.444	0.027	1.152
GAMLA	$\hat{\lambda}^{min}$	0.983	0.256	1.203
	$\hat{\lambda}^{1se}$	0.594	0.004	1.129
GAMA	$\alpha = 0.05$	0.938	0.057	1.148
	$\alpha = 0.01$	0.856	0.013	1.117
GAMSEL	$\hat{\lambda}^{min}$	0.853	0.546	1.164
	$\hat{\lambda}^{1se}$	0.286	0.100	1.203
OLS				1.471
Random Forest				1.185
XGBoost				1.216

Note: Potency and gauge are not reported for OLS, random forest and XGBoost because these models do not include variable selection. The results displayed correspond to average values of the criteria over 1000 replications.

Table 1 reports the average values of potency, gauge and MSE over all replications for the different models studied. The results suggest that the variable

selection of linear models augmented by parametric functions is inconsistent. Indeed, these models lead to a low value of potency due to the inconsistent estimation of parametric functions. While the highest percentage of relevant variables selected for these models is only equal to 60.4%, the smallest percentage drops to 2%. However, despite these low levels of potency, these models lead to relatively low MSE. This result implies that relevant interaction effects not included in the model are triangulated by combinations of other irrelevant variables, artificially increasing the predictive power of the model. Although it is not detrimental in terms of predictive performance, it is a major issue in terms of interpretability because it can lead users to incorrect recommendations. Indeed, instead of identifying predictive variables that truly explain the dependent variable, users can identify combinations of irrelevant variables recovering the true effects of relevant variables, leading to potentially misleading recommendations. Regarding gauge levels, the values obtained are decent although lasso leads to a high number of irrelevant interactions being included in the model for the $\hat{\lambda}^{min}$ rule of thumb. Moreover, these models are outperformed by GAM-based models. Despite leading to lower MSE values than traditional OLS and competing with the random forest and XGBoost, the parametric functions fail to capture non-linearities captured by the non-parametric functions of the GAMLA and GAMA models.

The results also highlight the importance of the double residual approach of Robinson (1988). Indeed, the GAMLA* and GAMA* models select even fewer relevant variables than the lasso and autometrics methods based on augmented linear models. At best, only half of the relevant interactions are included in these models. In comparison, the highest percentage of relevant variables included in the GAMLA and GAMA models is 98.3%, which is very close to the optimal potency value. These results imply that the double residual approach allows us to correctly estimate interaction effects, leading to consistent variable selection. Moreover, GAMLA and GAMA also achieve high predictive performance. Indeed, the MSEs associated with GAMLA and GAMA are lower than those of all other models, including the benchmark ML algorithms, with the smallest MSE being obtained for the GAMA model with $\alpha = 0.01$ as rule of thumb. Therefore, the results of the Monte Carlo simulation show that the partial linear models GAMA and GAMLA compete with sophisticated ML algorithms in terms of performance while leading to consistent identification of relevant and irrelevant variables.

The main difference between the GAMLA and GAMA models comes from the gauge level. On the one hand, the potency and gauge values are correlated for lasso: both values are high (low) with the rule of thumb $\hat{\lambda}^{min}$ ($\hat{\lambda}^{1se}$). Indeed, the highest potency value, 0.983, comes at the cost of a relatively high gauge level, 25.6%, whereas a gauge level close to 0 comes at the cost of the potency decreasing to 59.4%. Lasso thus leads to either potency values close to 1 (with $\hat{\lambda}^{min}$) or gauge levels close to 0 (with $\hat{\lambda}^{1se}$). On the other hand, autometrics leads to high

potency values while controlling for a low gauge level, unlike lasso. Specifically, the gauge level corresponds to the target size α of autometrics. Therefore, an autometrics user can choose the gauge desired while selecting a large percentage of relevant variables. Indeed, while the gauge levels are very close to the target sizes considered, $\alpha = 0.05$ or $\alpha = 0.01$, the potency levels remain high and are equal to 93.8% and 85.6%. For these reasons, we recommend using GAMA because it allows one to control the target size while also selecting a large percentage of relevant variables, unlike GAMLA. Finally, the weakness of GAMLA compared to the GAMA is also valid for GAMSEL. While leading to a competitive MSE close to those of GAMLA and GAMA, GAMSEL does not allow one to both control the target size and select a large percentage of relevant variables because it is based on a lasso penalisation.

We also display in Tables 5-7 in Appendix A the results obtained for three other Monte Carlo simulation setups. Specifically, we consider a DGP in which (i) covariates are correlated but functions g are linear, (ii) functions g are non-linear but covariates are independent, and (iii) functions g are linear and covariates are independent. For that purpose, we generate linear functions g such as $g_j(x_{i,j}) = \gamma_j x_{i,j}$, where $\gamma_j \neq 0$ for $j = 1, \ldots, 5$ and 0 otherwise, and simulate covariates as $x_{i,j} \sim \mathcal{N}(0, 1)$ for $j = 1, \ldots, p$ in the case of independence. The results obtained are consistent with those in Table 1: linear models augmented by parametric functions perform as well as GAMLA and GAMA when the relationship prevailing between y_i and x_i is purely linear, while GAMLA* and GAMA* perform similarly to GAMLA and GAMA when linear and non-linear terms are not correlated.

4 Empirical Application

In this section, we consider regression and classification problems to assess the potential of the GAMLA and GAMA models. For both applications, we evaluate the predictive performance of the GAMLA and GAMA models and their interpretability.

4.1 Regression Problem: Boston housing market

First, we illustrate the practical usefulness of the GAMLA and GAMA models based on a regression problem. For that purpose, we use the popular Boston housing market (Harrison Jr and Rubinfeld, 1978), which has already been considered in several contributions to the literature (Belsley et al., 2005; Michelucci and Venturini, 2021) as well as for the Kaggle competition "Boston Housing". Built by the U.S. Census Bureau, this dataset includes 506 instances and 13 explanatory variables on housing prices in the Boston area, two of which are qualitative. See Table 8 in Appendix A for a description of the variables in the dataset.

We compare the (1) GAMLA and (2) GAMA models to several benchmarks in the literature.[17] We consider three parametric models: (3) a simple linear regression including only linear terms, (4) a linear regression augmented by quadratic and cubic terms, and (5) a non-linear regression including linear, quadratic, cubic and interaction terms. We also consider (6) the GAM to compare the potential of parametric and non-parametric functions to capture non-linearities of covariates. We include in the comparison (7) lasso and (8) autometrics models based on the non-linear regression model including linear, quadratic, cubic and interaction terms to assess the importance of variable selection in a high-dimensional context.[18] We also implement the high-performing (9) random forest and (10) XGBoost algorithms and consider these models as benchmarks to evaluate the predictive performance of other approaches. Finally, we compare the GAMLA and GAMA models to the (11) penalised logistic tree regression model (PLTR) of Dumitrescu et al. (2021), as both types of model are designed to achieve the same objective.[19] Indeed, the PLTR is also intended to improve the predictive performance of traditional linear models by automatically capturing non-linearities. Specifically, it improves the predictive performance of linear models by including univariate and bivariate threshold effects obtained from short-depth decision trees and is penalised to control for the number of threshold effects included in the model. However, GAMLA and GAMA models and PLTR differ substantially in how non-linearities are captured. While GAMLA and GAMA are based on GAM non-parametric functions and interaction effects, the PLTR relies on univariate and bivariate threshold effects obtained from short-depth decision trees.

To evaluate the performance of these models, we use a 10-fold cross-validation approach based on the mean squared error (MSE), which is the benchmark performance measure for regression problems. For that purpose, we randomly divide the initial sample into 10 sub-samples of equal size and iteratively consider one sub-sample for prediction, while the nine other sub-samples are used to fit models. The MSE is then computed on the vector including all predictions obtained from each sub-sample. Moreover, we use the model confidence set (MCS) of Hansen et al. (2011) to identify models exhibiting significantly better predictions. Indeed, the MCS identifies the bucket of models that exhibit similar performance and are superior to the remaining models. To do so, we apply the MCS on the vector of squared errors obtained from the 10-fold cross-validation approach.

Finally, we analyse the interpretability of GAMLA and GAMA. However, mea-

[17] Similar to the Monte Carlo simulation, we consider a cubic basis for non-parametric functions of GAMLA and GAMA.

[18] We do not include in the comparison the GAMLA* and GAMA* methods because we demonstrated in Section 3 that these approaches lead to inconsistent variable selection in the presence of non-linear relationships prevailing between the dependent and predictive variables.

[19] Initially proposed for credit scoring applications, the PLTR can be extended to regression problems.

suring the interpretability of a model is difficult, and there is currently no consensus on the real definition of an "interpretable model" (Molnar, 2019). For that reason, we consider two criteria to measure the interpretability of GAMLA and GAMA that correspond to the function and human-level evaluation of interpretability proposed by Doshi-Velez and Kim (2017).[20] On the one hand, we assess the parsimony of the models and compute the number of interaction effects selected based on the same vector including all predictions used to compute the MSE. The idea of this quantitative criterion is that the fewer the number of variables involved in a prediction, the easier it is for a user to understand the determinants of predictions. On the other hand, we represent smooth functions and marginal effects of GAMA. These graphical representations can thus help the user to understand the effects detected by the model for each explanatory variable to clearly explain how predictions are obtained.

Table 2 displays the results obtained from the 10-fold cross-validation. The results suggest that (3 - 5) parametric linear models yield the worst performance of all models. Compared to the state-of-the-art (9) random forest and (10) XGBoost algorithms, the MSEs of the linear models are almost twice larger, even when augmented by parametric functions. Similarly, the performance of the (7) LASSO and (8) AM models is relatively poor compared to that of random forest and XGBoost, despite being better than that of linear models. These results confirm that including parametric functions in linear models fails to substantially improve their performance (Dumitrescu et al., 2021). In contrast, non-parametric functions of (6) GAM accurately capture non-linearities, with the MSE of GAM being closer to that of sophisticated ML algorithms than linear models. However, non-parametric functions are insufficient to achieve the performance of these high-performing algorithms, as linear models and GAM are not included in the subset of outperforming models identified by the MCS for a risk level $\alpha = 10\%$. These results thus highlight the importance of feature interactions for prediction, which can also be observed through the high performance of the (1) GAMLA and (2) GAMA models. Indeed, the combination of non-parametric GAM functions and interaction effects lead our models to very high performance. GAMLA and GAMA lead to lower MSE than (11) the PLTR model and are also included in the subset of outperforming models, while the PLTR is rejected from this subset. These results imply that the univariate and bivariate threshold effects of the PLTR are insufficient to capture all non-linearities in the dataset, unlike the non-parametric GAM functions and interaction effects of our models. The performance of GAMLA and GAMA is also competitive with random forest and XGBoost, with the smallest MSE being observed for the GAMLA model associated with the $\hat{\lambda}^{1se}$ rule of thumb. The MCS results also highlight the high performance of GAMLA and GAMA because they are generally included in the subset of outperforming models, similar to

[20] See Doshi-Velez and Kim (2017) for a detailed description of the interpretability evaluation.

Table 2: Number of variables selected, MSE and MCS: Boston housing dataset

#	Model	Rule of thumb	Number of interactions selected	MSE	MCS P-value
(1)	GAMLA	$\hat{\lambda}^{min}$	51	10.235	0.857
	GAMLA	$\hat{\lambda}^{1se}$	20	9.594	1
(2)	GAMA	$\alpha = 0.05$	27	10.086	0.907
	GAMA	$\alpha = 0.01$	21	10.389	0.857
(3)	Linear OLS			23.938	< 0.001
(4)	Non-Linear OLS (X^2, X^3)			16.039	0.006
(5)	Non-Linear OLS $(X_j^2, X_j^3, X_j X_k)$			24.079	0.006
(6)	GAM			13.186	0.017
(7)	LASSO	$\hat{\lambda}^{min}$	69	14.698	0.096
	LASSO	$\hat{\lambda}^{1se}$	31	15.683	0.006
(8)	AM	$\alpha = 0.05$	32	14.881	0.006
	AM	$\alpha = 0.01$	29	15.467	0.006
(9)	Random Forest			10.008	0.931
(10)	XGBoost			9.729	0.931
(11)	PLTR		37	13.726	0.006

Note: Linear OLS corresponds to a standard linear model estimated by OLS. Non-linear OLS (X^2, X^3) represents a linear model augmented by quadratic and cubic terms, estimated by OLS. Non-linear OLS $(X_j^2, X_j^3, X_j X_k)$ corresponds to a linear model augmented by quadratic, cubic and interaction terms, estimated by OLS. The MCS is computed from the MSE, based on 10,000 replications and a block length equal to 1. The total number of interaction effects for this dataset is 78.

the state-of-the-art ML algorithms. The take-away message here is that parametric models can compete with sophisticated ML algorithms in terms of predictive performance, as long as the models are well specified. Therefore, it is not essential to use black boxes to reach high predictive performance: parametric models can fulfil the same objective and even surpass them.

GAMLA and GAMA are also interpretable, unlike sophisticated ML algorithms. First, Table 2 suggests that GAMLA and GAMA are parsimonious because few interactions are included in the models, except for one case. Of the $(13 \times 12)/2 = 78$ interaction effects, fewer than 30 were selected by these models, leading to relatively small decision sets.[21] Moreover, these interactions are also easily interpretable because they are introduced linearly in GAMLA and GAMA. Second, graphical representations of non-parametric functions of GAMLA and GAMA allow us to identify estimated non-linear relationships. For example, Figure 1 displays estimated non-parametric functions of GAMA associated with a $\alpha = 0.05$ target size.[22] The results suggest that while the effect of the variable Rm is almost linear, for some other variables, the effects are non-linear, such as Black and Dis, which exhibit partially linear effects, or Lstat that displays a quadratic effect. These graphical representations thus allow the user to understand the estimated effect of each explanatory variable, which is not possible with ML algorithms such as random forest or XGBoost.

Finally, Figure 2 displays the marginal effect of the variable Lstat obtained for GAMA with a $\alpha = 0.05$ target size. The solid curve represents the marginal effect of Lstat when taken individually and shows that the marginal effect increases with Lstat and is positive from Lstat = 5.87. However, as Lstat interacts with four other covariates (Age, Black, Dis and Tax), its marginal effect does not correspond to the solid curve but to a translation from this curve representing the interactive effects of Lstat with these four other covariates. For illustration purposes, we compute this translation by considering the $\kappa = (0.025, 0.500, 0.975)$ quantiles of these four covariates. The marginal effects obtained are represented by the dashed and dotted curves. The results show that the marginal effect of Lstat simply corresponds to a decrease from the marginal effect of the associated smooth function.[23] These results highlight the ease of interpretation of the marginal effects of GAMA, which is an important added value of our new class of model to measure the importance of predictive variables.

[21] The fact that more than 20 interactions have been selected highlights the importance of interaction effects for the predictive performance of the models, as shown previously with the differences between GAM and GAMLA and GAMA. These interactions are the price to pay to achieve high predictive performance.

[22] Table 10 in Appendix A displays the estimation results of non-parametric functions of GAMA associated with a $\alpha = 0.05$ target size.

[23] The translation is equal to $-0.27, -1.18$ and -1.46 for $\kappa = (0.0.25, 0.500, 0.975)$, respectively.

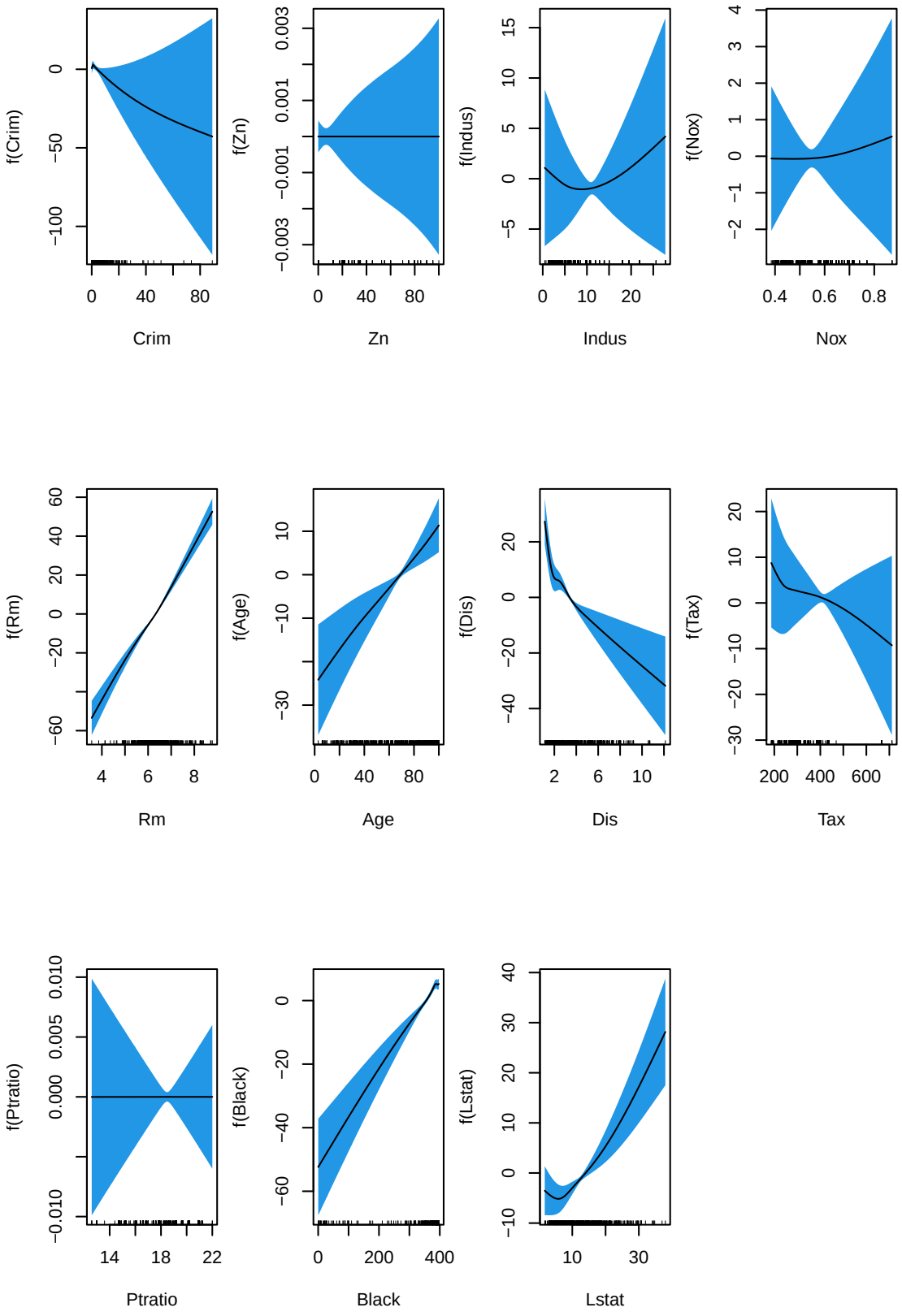

Figure 1: Non-parametric functions estimated for GAMA associated with the $\alpha = 0.05$ target size: Boston housing dataset

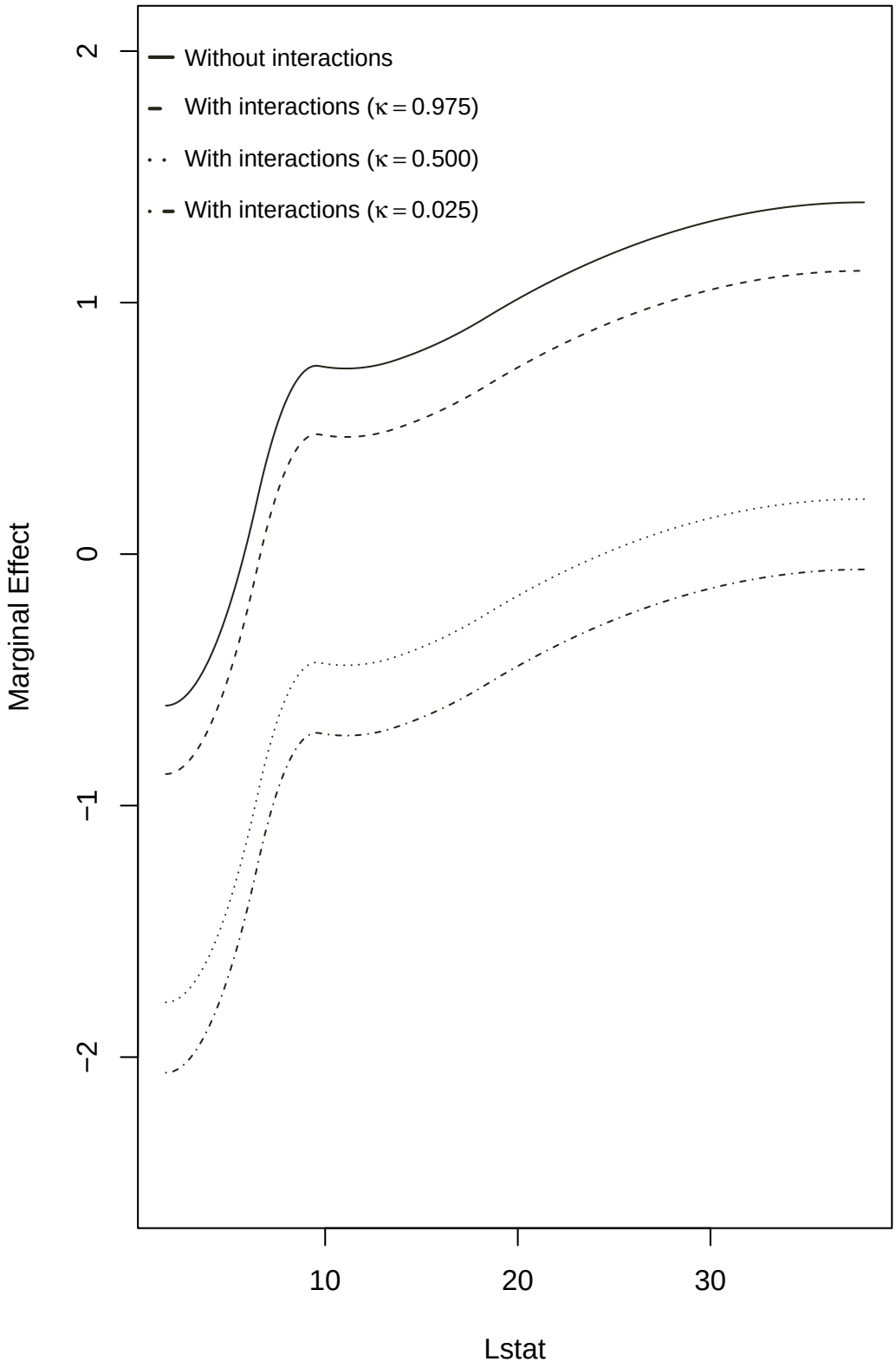

Figure 2: Marginal Effect of Lstat (with and without interactions) for GAMA associated with the $\alpha = 0.05$ target size: Boston housing dataset

4.2 Classification Problem: Credit scoring

Second, we consider a credit scoring application. The goal of credit scoring is to predict customer default and is based on the estimation of customers' default probability. To this end, we use the "Credit Card" dataset (Greene, 2003) that has also been used for Kaggle competitions. The dataset includes 1319 observations, and the dependent variable corresponds to the acceptation (1) and rejection (0) of customers' credit card applications. To explain the application decision, the dataset involves 11 explanatory variables, two of which are qualitative, and one continuous variable taking only two values. See Table 9 in Appendix A for a description of the variables in the dataset.

We compare the same models as previously considered in the Boston housing application. However, (1) GAMLA, (2) GAMA, (3 - 5) OLS, (6) GAM, (7) LASSO and (8) AM are linear probability models instead of traditional linear models because the dependent variable is binary.[24] To assess the performance of the models, we rely on a 10-fold cross-validation approach based on the area under the ROC curve (AUC), which is the benchmark performance measure for classification problems. The AUC measures the link between the false and true positive rates over every possible threshold between 0 and 1. Specifically, the AUC represents the probability that the occurrence of a random bad application is higher than that of a random good application. Moreover, to assess whether the AUC difference between two models is significant, we use the pairwise AUC test used in Candelon et al. (2012). Finally, we also compute the number of variables selected and represent estimated non-parametric functions of GAMLA and GAMA to study their interpretability.

Table 3 displays the number of variables selected and the AUC for each model obtained for the credit card dataset. Similarly to the Boston housing dataset, (3 - 5) parametric linear models lead to lower predictive performance, and parametric functions fail to accurately capture non-linearities. Indeed, the AUC of the parametric linear models, (7) LASSO and (8) AM models are worse than those of (9) random forest and (10) XGBoost. However, non-parametric functions of (6) GAM accurately capture non-linearities and lead to highly performing models. Indeed, GAM, (1) GAMLA and (2) GAMA achieve similar performance to random forest, XGBoost and (11) PLTR. These results highlight the potential of non-parametric GAM functions to capture non-linearities because the PLTR, random forest and XGBoost have been identified as benchmark models for credit scoring applications (Lessmann et al., 2015; Grennepois et al., 2018; Dumitrescu et al., 2021; Gunnarsson et al., 2021). Moreover, the results also suggest that the predictive performance of these models mostly comes from non-linearities of covariates rather than interaction effects. Indeed, unlike the Boston housing application, GAM leads to similar

[24]The PLTR is estimated by a logit regression as proposed in Dumitrescu et al. (2021).

Table 3: Number of variables selected and AUC: Credit card dataset

#	Model	Rule of thumb	Number of interactions selected	AUC
(1)	GAMLA	$\hat{\lambda}^{min}$	12	0.995
	GAMLA	$\hat{\lambda}^{1se}$	0	0.995
(2)	GAMA	$\alpha = 0.05$	3	0.995
	GAMA	$\alpha = 0.01$	2	0.995
(3)	Linear OLS			0.924
(4)	Non-Linear OLS (X^2, X^3)			0.967
(5)	Non-Linear OLS $(X_j^2, X_j^3, X_j X_k)$			0.988
(6)	GAM			0.995
(7)	LASSO	$\hat{\lambda}^{min}$	35	0.964
	LASSO	$\hat{\lambda}^{1se}$	27	0.963
(8)	AM	$\alpha = 0.05$	9	0.982
	AM	$\alpha = 0.01$	5	0.985
(9)	Random Forest			0.995
(10)	XGBoost			0.996
(11)	PLTR		7	0.996

Note: Linear OLS corresponds to a standard linear probability model. Non-linear OLS (X^2, X^3) represents a linear probability model augmented by quadratic and cubic terms. Non-linear OLS $(X_j^2, X_j^3, X_j X_k)$ corresponds to a linear probability model augmented by quadratic, cubic and interaction terms. The total number of interaction effects for this dataset is 55.

Table 4: P-values of pairwise bilateral AUC tests: Credit card dataset

	XGBoost	Random Forest	GAM	GAMLA ($\hat{\lambda}^{min}$)	GAMLA ($\hat{\lambda}^{1se}$)	GAMA ($\alpha = 0.05$)	GAMA ($\alpha = 0.01$)	PLTR
XGBoost	-							
Random Forest	0.511	-						
GAM	0.749	0.679	-					
GAMLA ($\hat{\lambda}^{min}$)	0.676	0.752	0.773	-				
GAMLA ($\hat{\lambda}^{1se}$)	0.799	0.637	0.753	0.658	-			
GAMA ($\alpha = 0.05$)	0.847	0.592	0.643	0.323	0.769	-		
GAMA ($\alpha = 0.01$)	0.942	0.508	0.441	0.237	0.485	0.541	-	
PLTR	0.415	0.129	0.218	0.163	0.241	0.259	0.350	-

Note: This table displays the p-values of pairwise bilateral test of AUC (Candelon et al., 2012) for the XGBoost, random forest, GAM, GAMLA, GAMA and PLTR models. P-values associated with the OLS, LASSO and AM models are not reported here but are all inferior to 0.001.

performance to GAMLA, GAMA, random forest and XGBoost, implying that the importance of interaction effects is negligible for this dataset. To confirm this result, we display in Table 4 the p-values of a pairwise bilateral test of the AUC, which tests whether the difference between the AUCs of two competing models is significant. For the sake of clarity, results related to the OLS, LASSO and AM models are not displayed, but we find that these models are rejected because their p-values are all inferior to 0.001, which confirms that parametric functions fail to substantially improve the predictive performance of parametric models.[25] Regarding XGBoost, random forest, GAM, GAMLA, GAMA and PLTR, the results show that these models lead to similar AUCs because none of the p-values are lower than the risk level $\alpha = 10\%$. This result is also confirmed by the small number of interaction effects selected by the GAMLA and GAMA models, i.e., fewer than 10 interactions except for one case. The take-away message here is that interaction effects are insignificant in this dataset, which explains why GAM performs as well as more sophisticated approaches.

Finally, GAMLA and GAMA still appear to be easily interpretable. The models select even fewer interaction effects than in the previous application, and these interactions remain interpretable due to the linear assumption on the interaction variables. Moreover, Figure 3 displays estimated non-linearities for GAMA associated with a $\alpha = 0.05$ target size.[26] Notable among the results, a quadratic effect can be observed for the variable Active, a partially linear effect with a threshold around 3 for the variable Income, and a highly non-linear and particular effect for the variable Share.

5 Conclusion

In the wake of the growing use of machine learning (ML) algorithms in economics, interpretability has returned to the heart of the literature. Despite their high predictive performance, ML algorithms are black boxes, which leads to uninterpretable models. The opacity of ML algorithms has raised concerns from practitioners and regulators (Bracke et al., 2019; ACPR, 2020; EBA, 2020; EC, 2020) and limits their use in industry. Two types of approach are currently investigated in the literature to redirect the focus to the interpretability of models. On the one hand, a first strand of the literature proposes model-agnostic approaches to improve the interpretability of black-box models. On the other hand, a second strand instead designs inherently interpretable models. While the first group of contributions has received considerably more attention in the literature, these approaches can

[25] See the footnote of Table 4.

[26] Table 11 in Appendix A displays estimation results of non-parametric functions of GAMA associated with a $\alpha = 0.05$ target size.

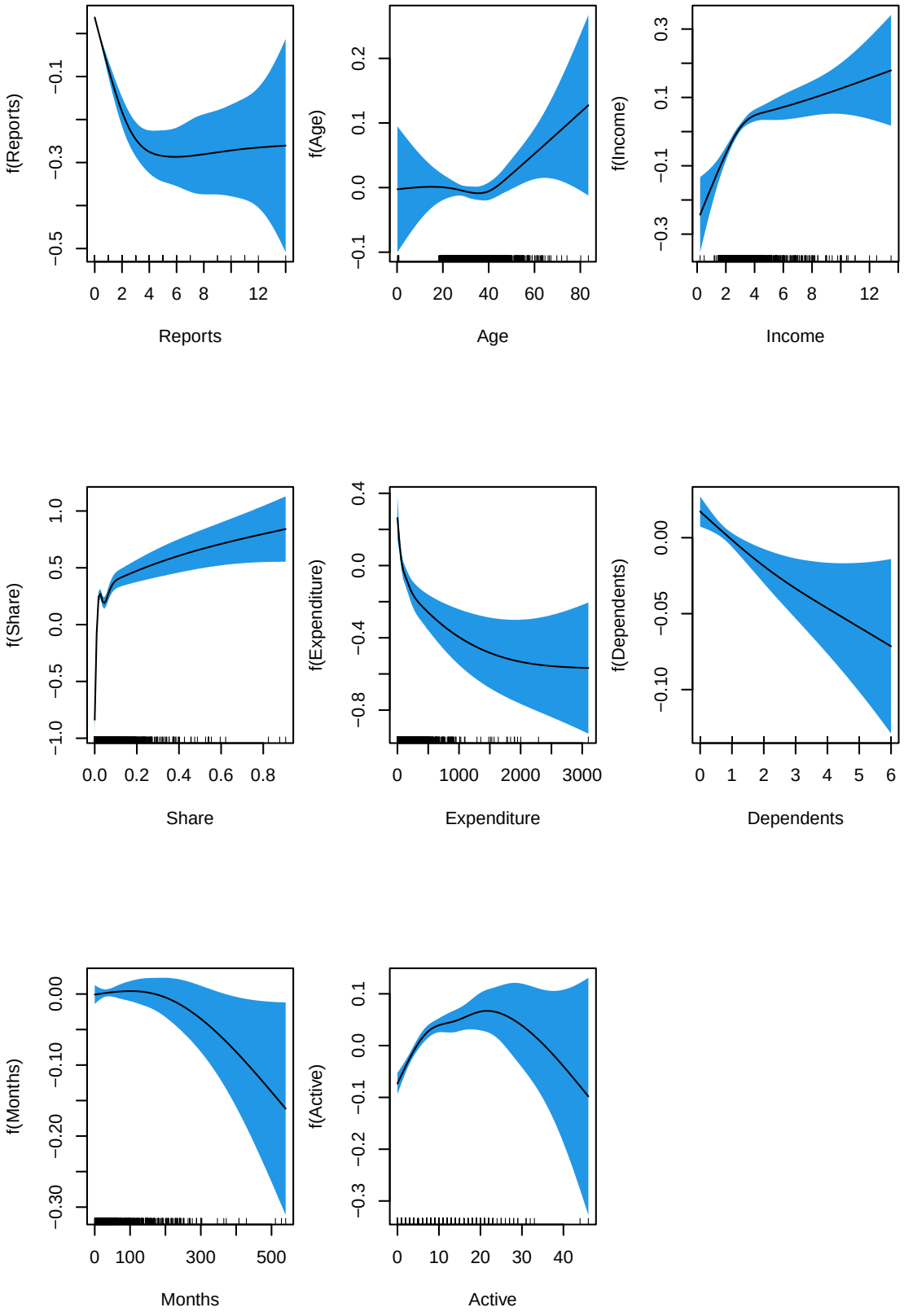

Figure 3: Non-parametric functions estimated for GAMA associated with the $\alpha = 0.05$ target size: credit card dataset

potentially mislead users in high-stakes decisions (Rudin, 2019).

Against this background, we propose in this paper a class of partial linear models that are inherently interpretable that also competes with sophisticated ML algorithms in terms of predictive performance. Specifically, this article introduces GAM-lasso (GAMLA) and GAM-autometrics (GAMA) models, which combine parametric and non-parametric functions to accurately capture linearities and non-linearities prevailing between dependent and explanatory variables, into a variable selection procedure to control for overfitting issues. However, the presence of linear and non-linear terms in the models can lead to inconsistent variable selection. To solve this issue, we propose to rely on the double residual approach of Robinson (1988) that allows one to recover the consistency of variable selection. Monte Carlo simulation experiments show that GAMLA and GAMA can compete with benchmark ML algorithms in terms of predictive performance. Moreover, the results highlight the importance of the double residual approach as part of GAMLA and GAMA. We also recommend using GAMA because it allows the researcher to control the gauge level while leading to satisfying potency values, unlike GAMLA.

In the empirical application, we illustrate the predictive performance and interpretability of GAMLA and GAMA from regression and classification problems. Specifically, we compare GAMLA and GAMA to several other benchmark models in the literature by considering traditional measures of performance and inference procedures. The results show that GAMLA and GAMA outperform parametric models and parametric models augmented by quadratic, cubic and interaction effects. Moreover, the results also suggest that the performance of our new models is not significantly different from that of benchmark ML algorithms and is even better in some cases. We also illustrate the interpretability of GAMLA and GAMA and show that the variable selection leads to parsimonious models while graphical representations of smooth functions and marginal effects allow a simple understanding of the relationships identified by the models.

Finally, we show in this paper that it is possible to design inherently interpretable models capable of competing with sophisticated ML algorithms in terms of predictive performance, and we advocate for more work in this direction, similarly to Rudin (2019).

References

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

Apley, D. W. and Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(4):1059–1086.

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

Barocas, S., Hardt, M., and Narayanan, A. (2018). Fairness and machine learning. fairmlbook. org, 2019.

Belsley, D. A., Kuh, E., and Welsch, R. E. (2005). *Regression diagnostics: Identifying influential data and sources of collinearity*, volume 571. John Wiley & Sons.

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

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and regression trees*. Routledge.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (2017). *Classification and regression trees*. Routledge.

Candelon, B., Dumitrescu, E.-I., and 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.

Castle, J. L., Clements, M. P., and Hendry, D. F. (2013). Forecasting by factors, by variables, by both or neither? *Journal of Econometrics*, 177(2):305–319.

Castle, J. L., Doornik, J. A., and Hendry, D. F. (2011). Evaluating automatic model selection. *Journal of Time Series Econometrics*, 3(1).

Castle, J. L. and Hendry, D. F. (2010). A low-dimension portmanteau test for non-linearity. *Journal of econometrics*, 158(2):231–245.

Charpentier, A., Flachaire, E., and Ly, A. (2018). Econometrics and machine learning. *Economie et Statistique*, 505(1):147–169.

Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., et al. (2015). Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1–4.

Chouldechova, A. and Hastie, T. (2015). Generalized additive model selection. *arXiv preprint arXiv:1506.03850*.

Desai, V. S., Crook, J. N., and 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.

Doornik, J. A. et al. (2009). Autometrics. *Castle, and Shephard (2009)*, pages 88–121.

Doornik, J. A. and Hansen, H. (2008). An omnibus test for univariate and multivariate normality. *Oxford bulletin of economics and statistics*, 70:927–939.

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

Dumitrescu, E., Hue, S., Hurlin, C., and Tokpavi, S. (2021). Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects. *European Journal of Operational Research*.

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.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica*, pages 987–1007.

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

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232.

Frisch, R. and Waugh, F. V. (1933). Partial time regressions as compared with individual trends. *Econometrica*, pages 387–401.

Ghosal, I. and Kormaksson, M. (2019). The plsmselect package. https://cran.r-project.org/web/packages/plsmselect/vignettes/plsmselect.html.

Godfrey, L. G. (1978). Testing for higher order serial correlation in regression equations when the regressors include lagged dependent variables. *Econometrica*, pages 1303–1310.

Greene, W. (2003). Econometric analysis 5th edition pearson education inc.

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

Gunnarsson, B. R., Vanden Broucke, S., Baesens, B., Óskarsdóttir, M., and Lemahieu, W. (2021). Deep learning for credit scoring: Do or don't? *European Journal of Operational Research*, 295(1):292–305.

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

Harrison Jr, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air.

Hastie, T. J. and Tibshirani, R. J. (2017). *Generalized additive models*. Routledge.

Hendry, D. F. (2000). *Econometrics: alchemy or science?: essays in econometric methodology*. Oxford University Press.

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

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

Hurlin, C., Pérignon, C., and Saurin, S. (2021). The fairness of credit scoring models. *Available at SSRN 3785882*.

Kozodoi, N., Jacob, J., and Lessmann, S. (2021). Fairness in credit scoring: Assessment, implementation and profit implications. *European Journal of Operational Research*.

Lessmann, S., Baesens, B., Seow, H.-V., and 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.

Li, Q. and Racine, J. S. (2007). *Nonparametric econometrics: theory and practice*. Princeton University Press.

Lovell, M. C. (1963). Seasonal adjustment of economic time series and multiple regression analysis. *Journal of the American Statistical Association*, 58(304):993–1010.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pages 4768–4777.

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

Michelucci, U. and Venturini, F. (2021). Estimating neural network's performance with bootstrap: A tutorial. *Machine Learning and Knowledge Extraction*, 3(2):357–373.

Molnar, C. (2019). Interpretable machine learning: A guide for making black box models explainable. published online.

Molnar, C., Casalicchio, G., and Bischl, B. (2020). Interpretable machine learning–a brief history, state-of-the-art and challenges. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 417–431. Springer.

Paleologo, G., Elisseeff, A., and Antonini, G. (2010). Subagging for credit scoring models. *European journal of operational research*, 201(2):490–499.

Ramsey, J. B. (1969). Tests for specification errors in classical linear least-squares regression analysis. *Journal of the Royal Statistical Society: Series B (Methodological)*, 31(2):350–371.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Robinson, P. M. (1988). Root-n-consistent semiparametric regression. *Econometrica:*, pages 931–954.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., and Zhong, C. (2021). Interpretable machine learning: Fundamental principles and 10 grand challenges. *arXiv preprint arXiv:2103.11251*.

Rudin, C. and Radin, J. (2019). Why are we using black box models in AI when we don't need to? A lesson from an explainable AI competition. *Harvard Data Science Review*, 1(2).

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

Varian, H. R. (2014). Big data: New tricks for econometrics. *Journal of Economic Perspectives*, 28(2):3–28.

White, H. (1980). A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica*, pages 817–838.

A Additional Tables

Table 5: Comparison of potency, gauge and MSE under non-linear effects and uncorrelated covariates

Model	Rule of thumb	Potency	Gauge	MSE
LASSO	$\hat{\lambda}^{min}$	1	0.365	1.107
	$\hat{\lambda}^{1se}$	0.997	0.038	1.149
AM	$\alpha = 0.05$	1	0.049	1.104
	$\alpha = 0.01$	0.999	0.011	1.098
GAMLA*	$\hat{\lambda}^{min}$	1	0.249	1.092
	$\hat{\lambda}^{1se}$	0.991	0.006	1.064
GAMA*	$\alpha = 0.05$	1	0.051	1.073
	$\alpha = 0.01$	1	0.011	1.064
GAMLA	$\hat{\lambda}^{min}$	1	0.270	1.094
	$\hat{\lambda}^{1se}$	0.992	0.008	1.064
GAMA	$\alpha = 0.05$	1	0.055	1.074
	$\alpha = 0.01$	1	0.012	1.064
GAMSEL	$\hat{\lambda}^{min}$	1	0.474	1.110
	$\hat{\lambda}^{1se}$	0.999	0.060	1.144
OLS				1.156
Random Forest				1.203
XGBoost				1.291

Note: Potency and gauge are not reported for OLS, random forest and XGBoost because these models do not include variable selection. The results displayed correspond to average values of criteria over 1000 replications.

Table 6: Comparison of potency, gauge and MSE under linear effects and correlated covariates

Model	Rule of thumb	Potency	Gauge	MSE
LASSO	$\hat{\lambda}^{min}$	1	0.643	1.058
	$\hat{\lambda}^{1se}$	0.990	0.459	1.101
AM	$\alpha = 0.05$	1	0.048	1.026
	$\alpha = 0.01$	1	0.011	1.015
GAMLA*	$\hat{\lambda}^{min}$	1	0.753	1.077
	$\hat{\lambda}^{1se}$	0.955	0.368	1.073
GAMA*	$\alpha = 0.05$	0.998	0.211	1.056
	$\alpha = 0.01$	0.973	0.152	1.057
GAMLA	$\hat{\lambda}^{min}$	1	0.273	1.068
	$\hat{\lambda}^{1se}$	0.993	0.008	1.038
GAMA	$\alpha = 0.05$	1	0.056	1.048
	$\alpha = 0.01$	1	0.013	1.039
GAMSEL	$\hat{\lambda}^{min}$	1	0.710	1.072
	$\hat{\lambda}^{1se}$	0.987	0.515	1.108
OLS				1.091
Random Forest				1.412
XGBoost				1.213

Note: Potency and gauge are not reported for OLS, random forest and XGBoost because these models do not include variable selection. The results displayed correspond to average values of criteria over 1000 replications.

Table 7: Comparison of potency, gauge and MSE under linear effects and uncorrelated covariates

Model	Rule of thumb	Potency	Gauge	MSE
LASSO	$\hat{\lambda}^{min}$	1	0.307	1.037
	$\hat{\lambda}^{1se}$	1	0.032	1.077
AM	$\alpha = 0.05$	1	0.052	1.034
	$\alpha = 0.01$	1	0.011	1.020
GAMLA*	$\hat{\lambda}^{min}$	01	0.252	1.066
	$\hat{\lambda}^{1se}$	0.994	0.006	1.037
GAMA*	$\alpha = 0.05$	1	0.052	1.047
	$\alpha = 0.01$	1	0.010	1.038
GAMLA	$\hat{\lambda}^{min}$	1	0.267	1.067
	$\hat{\lambda}^{1se}$	0.995	0.008	1.037
GAMA	$\alpha = 0.05$	1	0.055	1.048
	$\alpha = 0.01$	1	0.011	1.038
GAMSEL	$\hat{\lambda}^{min}$	1	0.373	1.033
	$\hat{\lambda}^{1se}$	1	0.045	1.064
OLS				1.093
Random Forest				1.181
XGBoost				1.218

Note: Potency and gauge are not reported for OLS, random forest and XGBoost because these models do not include variable selection. The results displayed correspond to average values of criteria over 1000 replications.

Table 8: Description of the variables in the Boston dataset

Variable	Description
Medv	Median value of owner-occupied homes in $1000s
Crim	Per capita crime rate by town
Zn	Proportion of residential land zoned for lots over 25,000 square feet
Indus	Proportion of non-retail business acres per town
Chas	Charles River dummy variable (= 1 if tract bounds river, 0 otherwise)
Nox	Nitric oxides concentration (parts per 10 million)
Rm	Average number of rooms per dwelling
Age	Proportion of owner-occupied units built prior to 1940
Dis	Weighted distances to five Boston employment centres
Rad	Index of accessibility to radial highways
Tax	Full-value property-tax rate per $10,000
Ptratio	Pupil-teacher ratio by town
Black	$1000(B_k - 0.63)^2$, where B_k is the proportion of black persons by town
Lstat	% Lower status of the population

Note: See Harrison Jr and Rubinfeld (1978) for more details on the dataset.

Table 9: Description of the variables in the credit card dataset

Variable	Description
Card	Dummy variable: 1 if application for credit card accepted, 0 if not
Reports	Number of major derogatory reports
Age	Age in years plus twelfths of a year
Income	Yearly income (in USD 10,000)
Share	Ratio of monthly credit card expenditure to yearly income
Expenditure	Average monthly credit card expenditure
Owner	Dummy variable: 1 if owns their home, 0 if rent
Selfemp	Dummy variable: 1 if self employed, 0 if not
Dependents	Number of dependents
Months	Months living at current address
Majorcards	Number of major credit cards held (0 or 1)
Active	Number of active credit accounts

Note: See Greene (2003) for more details on the dataset.

Table 10: Estimation results of non-parametric functions of GAMA associated with the $\alpha = 0.05$ target size: Boston housing dataset

Variable	Edf	Df	F-stat	P-value
Crim	4.488	5	6.870	< 0.001
Zn	0.000	5	0.000	0.681
Indus	1.840	5	2.308	0.001
Nox	0.235	5	0.067	0.200
Rm	3.997	5	55.557	< 0.001
Age	4.164	5	3.271	0.002
Dis	4.974	5	26.247	< 0.001
Tax	4.302	5	4.201	< 0.001
Ptratio	0.000	5	0.000	0.458
Black	4.705	5	11.481	< 0.001
Lstat	4.551	5	15.263	< 0.001

Note: This table displays estimation results of non-parametric functions of GAMA associated with the $\alpha = 0.05$ target size for the Boston housing dataset.

Table 11: Estimation results of non-parametric functions of GAMA associated with the $\alpha = 0.05$ target size: Credit card dataset

Variable	Edf	Df	F-stat	P-value
Reports	3.222	5	35.754	< 0.001
Age	2.715	5	1.595	0.023
Income	3.334	5	9.451	< 0.001
Share	5.000	5	84.070	< 0.001
Expenditure	4.770	5	5.821	< 0.001
Dependents	1.120	5	2.425	< 0.001
Months	2.070	5	0.990	0.067
Active	3.680	5	10.915	< 0.001

Note: This table displays estimation results of non-parametric functions of GAMA associated with the $\alpha = 0.05$ target size for the credit card dataset.