European Journal of Operational Research 299 (2022) 1045–1054

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Analytics, Computational Intelligence and Information Management

On sparse optimal regression trees

Rafael Blanquero[a], Emilio Carrizosa[a], Cristina Molero-Río[a,*], Dolores Romero Morales[b]

[a] Instituto de Matemáticas de la Universidad de Sevilla (IMUS), Seville, Spain
[b] Copenhagen Business School (CBS), Frederiksberg, Denmark

ARTICLE INFO

Article history:
Received 5 May 2020
Accepted 14 December 2021
Available online 18 December 2021

Keywords:
Machine learning
Classification and regression trees
Optimal regression trees
Sparsity
Nonlinear programming

ABSTRACT

In this paper, we model an optimal regression tree through a continuous optimization problem, where a compromise between prediction accuracy and both types of sparsity, namely local and global, is sought. Our approach can accommodate important desirable properties for the regression task, such as cost-sensitivity and fairness. Thanks to the smoothness of the predictions, we can derive local explanations on the continuous predictor variables. The computational experience reported shows the outperformance of our approach in terms of prediction accuracy against standard benchmark regression methods such as CART, OLS and LASSO. Moreover, the scalability of our approach with respect to the size of the training sample is illustrated.

1. Introduction

Regression Analysis is one of the most used tasks in Statistics and Machine Learning (Hastie, Tibshirani, & Friedman, 2009). The classic linear regression is known to be outperformed by many proposals that apply non-linear techniques, such as tree-based methods, which are the focus of this paper. Tree-based methods (Chikalov, Hussain, & Moshkov, 2018; Hu, Rudin, & Seltzer, 2019; Yang, Liu, Tsoka, & Papageorgiou, 2017) are appealing due to their learning performance and, since they are rule-based, seen as interpretable (Athey, 2018; Baesens, Setiono, Mues, & Vanthienen, 2003; Carrizosa, Martín-Barragán, & Romero Morales, 2011; Freitas, 2014; Goodman & Flaxman, 2017; Jung, Concannon, Shroff, Goel, & Goldstein, 2017; Martens, Baesens, Van Gestel, & Vanthienen, 2007; Martín-Barragán, Lillo, & Romo, 2014; Ridgeway, 2013; Ustun & Rudin, 2016).

Building optimal decision trees is an NP-complete task (Hyafil & Rivest, 1976). For this reason, greedy heuristic procedures such as CART (Breiman, Friedman, Stone, & Olshen, 1984) have been proposed, yielding suboptimal trees instead. Even though some attempts (Bennett & Blue, 1996) were made in the past, the latest advances in both computer performance and Mathematical Optimization have led to a growing research on building such optimal decision trees (Bertsimas, Dunn, & Paschalidis, 2017; Better, Glover, & Samorani, 2010; Blanquero, Carrizosa, Molero-Río, & Romero Morales, 2020; Blanquero, Carrizosa, Molero-Río, & Romero Morales, 2021a; Dunn, 2018; Firat, Crognier, Gabor, Hurkens, & Zhang, 2019; Günlük, Kalagnanam, Li, Menickelly, & Scheinberg, 2021; Narodytska, Ignatiev, Pereira, Marques-Silva, & RAS, 2018; Verwer & Zhang, 2017; 2019). The reader is referred to (Carrizosa, Molero-Río, & Romero Morales, 2021) for a review on this topic.

The modeling of a decision tree via Mathematical Optimization yields, in general, an improvement in prediction accuracy with respect to traditional approaches, but, equally important, it allows the user to easily deal with desirable properties in Machine Learning that globally involve all the decision rules along the tree. Such is the case of global sparsity (Tibshirani, Wainwright, & Hastie, 2015). While heuristic procedures, such as CART, or more sophisticated tree-based approaches, such as Random Forest (RF) (Biau & Scornet, 2016; Breiman, 2001; Fernández-Delgado, Cernadas, Barro, & Amorim, 2014; Genuer, Poggi, Tuleau-Malot, & Villa-Vialaneix, 2017), easily control local sparsity, that is, the number of predictor variables to be used at each splitting rule, they find it hard to control global sparsity, that is, the number of predictor variables to be used across the tree (Deng & Runger, 2012; 2013; Ruggieri, 2019). This is not the case for approaches based on mathematical optimization which are flexible enough to model this objective directly (Bertsimas et al., 2017; Blanquero et al., 2020; Dunn, 2018; Firat et al., 2019; Verwer & Zhang, 2017), either with a LASSO term, or by adding binary decision variables and additional constraints. In this paper, we tackle this issue and propose the Sparse Optimal Randomized Regression Tree (S-ORRT). An S-ORRT seeks a

[*] Corresponding author.
E-mail addresses: rblanquero@us.es (R. Blanquero), ecarrizosa@us.es (E. Carrizosa), mmolero@us.es (C. Molero-Río), drm.eco@cbs.dk (D.R. Morales).

https://doi.org/10.1016/j.ejor.2021.12.022

good tradeoff between both prediction accuracy and both types of sparsity, obtained by minimizing the mean squared error over the training sample, as customary in Regression Analysis, plus two regularization terms. Other global desirable properties that one may care for include the modeling of cost-sensitivity (Günlük et al., 2021) or fairness (Aghaei, Azizi, & Vayanos, 2019) constraints, with the aim to protect critical groups or to avoid the discrimination of groups that share sensitive features, respectively.

Optimal regression trees (Bertsimas et al., 2017; Dunn, 2018; Verwer & Zhang, 2017) have been recently formulated using mixed-integer models. These models include an integer decision variable for each individual, as well as one for each predictor variable. The resulting combinatorial framework hinders the tractability of the problem when the dimensionality of the data grows, yielding a significant computational effort even for small data sets. Local-search strategies have been proposed to alleviate the computational burden of these procedures (Dunn, 2018), however they cannot control global desirable properties. Our approach considers a continuous optimization model instead, where there are no decision variables directly relating to the individuals, making it scalable with respect to the training sample. This is achieved through (i) the inclusion of a continuous cumulative density function F at each branch node that smoothens the transition from the left child node to the right one, and (ii) the use of the ℓ_1 and ℓ_∞ norms to control local and global sparsity, respectively.

Thanks to the smoothness of our approach, the impact that continuous predictor variables have on the individual prediction, that is, local explanations (Lundberg et al., 2020; Lundberg & Lee, 2017; Molnar, Casalicchio, & Bischl, 2020; Ribeiro, Singh, & Guestrin, 2016), can be easily derived. For nonlinear models one can make use of generic post-hoc approaches to build local explanations, such as the so-called Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016). Instead, and as advocated by Rudin (2019), one can work with models that derive local explanations directly (Gevrey, Dimopoulos, & Lek, 2003), as we do.

The remainder of the paper is organized as follows. In Section 2, we introduce the S-ORRT and its mathematical formulation, as well as the modeling of desirable properties. Some theoretical properties of S-ORRT are discussed in Section 3. Technical proofs can be found in the Appendix. In Section 4, our computational experience is reported. We illustrate that S-ORRT outperforms the benchmark regression methods CART, OLS and LASSO in terms of prediction accuracy. Moreover, we show our ability to easily trade in prediction accuracy for a gain in local and global sparsity, as well as our favorable scalability with respect to the size of the training sample. Finally, conclusions and possible lines of future research are provided in Section 5.

2. Sparse optimal randomized regression trees

2.1. Introduction

Let $\mathcal{I}$ be a given set of individuals. Each individual $i \in \mathcal{I}$ has associated a pair $(\boldsymbol{x}_i, y_i)$, where $\boldsymbol{x}_i$ represents the p-dimensional vector of predictor variables of individual i, and $y_i \in \mathbb{R}$ indicates the value of the response variable.

A Sparse Optimal Randomized Regression Tree (S-ORRT) is an optimal binary regression tree of a given depth D, obtained by controlling simultaneously prediction accuracy and local and global sparsity. We briefly sketch here this randomized framework. For further details on the construction of optimal randomized trees, the reader is referred to (Blanquero et al., 2020; Blanquero et al., 2021a). Figure 1 shows the structure of an S-ORRT of depth $D = 2$. Unlike classic decision trees, oblique cuts, on which more than one predictor variable is involved, are implemented. S-ORRTs are modeled by means of a Non-Linear Continuous Optimization (NLCO) formulation. The usual deterministic yes/no rule at each branch node is replaced by a smoother rule: a probabilistic decision rule at each branch node, induced by a cumulative density function (CDF) F, is obtained. Therefore, the movements in S-ORRTs can be seen as randomized: at a given branch node of an S-ORRT, a random variable will be generated to indicate by which branch an individual has to continue. Since binary trees are built, the Bernoulli distribution is appropriate, whose probability of success will be determined by the value of this CDF, evaluated over the vector of predictor variables. More precisely, at a given branch node t of the tree, an individual with predictor variables $\boldsymbol{x}_i$ will go either to the left or to the right child nodes with probabilities $F\left(\frac{1}{p}\boldsymbol{a}_{\cdot t}^T\boldsymbol{x}_i - \mu_t\right)$ and $1 - F\left(\frac{1}{p}\boldsymbol{a}_{\cdot t}^T\boldsymbol{x}_i - \mu_t\right)$, respectively, where $\boldsymbol{a}_{\cdot t}$ and μ_t are decision variables of the optimization problem that needs to be solved to build the S-ORRT. In Fig. 1, p_{i1}, p_{i2}, p_{i3} and their complement to one denote such probabilities for the three branch nodes. With this, we have the probability of each individual in the sample falling into every leaf node. In Fig. 1, P_{i4}, P_{i5}, P_{i6} and P_{i7} denote such probabilities. To end, we need to define how S-ORRT makes predictions. First, S-ORRT associates linear predictions to each leaf node. Then, the estimated outcome value for each individual is defined as the summation of these linear predictions, weighted by the probability of belonging to the corresponding leaf node. This is denoted by φ_{i4}, φ_{i5}, φ_{i6} and φ_{i7} in Fig. 1.

The following notation is required:

Parameters	
D	depth of the binary tree,
p	number of predictor variables,
$\{(\boldsymbol{x}_i, y_i)\}_{i\in\mathcal{I}}$	training sample, where $\boldsymbol{x}_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$, with cardinality $\lvert\mathcal{I}\rvert$,
$F(\cdot)$	univariate continuously differentiable CDF, used to define the probabilities for an individual to go to the left or the right child node in the tree,
λ^L, λ^G	local and global sparsity regularization parameters.
Nodes	
τ_B	set of branch nodes,
τ_L	set of leaf nodes,
$N_L(t)$	set of ancestor nodes of leaf node t whose left branch takes part in the path from the root node to leaf node t, $t \in \tau_L$,
$N_R(t)$	set of ancestor nodes of leaf node t whose right branch takes part in the path from the root node to leaf node t, $t \in \tau_L$.
Decision variables	
$a_{jt} \in \mathbb{R}$	coefficient of predictor variable j in the oblique cut at branch node $t \in \tau_B$, or in the linear prediction at leaf node $t \in \tau_L$. The expressions $\boldsymbol{a}_B$ and $\boldsymbol{a}_L$ will denote the $p \times \lvert\tau_B\rvert$- and $p \times \lvert\tau_L\rvert$-matrices that involve these coefficients, respectively, $\boldsymbol{a}_B = \left(a_{jt}\right)_{j=1,\ldots,p,\ t\in\tau_B}$ and $\boldsymbol{a}_L = \left(a_{jt}\right)_{j=1,\ldots,p,\ t\in\tau_L}$. Let $\boldsymbol{a}$ denote the $p \times (\lvert\tau_B\rvert + \lvert\tau_L\rvert)$-matrix $\boldsymbol{a} = \left(a_{jt}\right)_{j=1,\ldots,p,\ t\in\tau_B\cup\tau_L} = (\boldsymbol{a}_B, \boldsymbol{a}_L)$. Both notations will be used interchangeably when needed. The expressions $\boldsymbol{a}_{j\cdot}$ and $\boldsymbol{a}_{\cdot t}$ will denote the jth row and the tth column of $\boldsymbol{a}$, respectively,
$\mu_t \in \mathbb{R}$	location parameter at branch node $t \in \tau_B$, or intercept of the linear prediction at leaf node $t \in \tau_L$. The expressions $\boldsymbol{\mu}_B$ and $\boldsymbol{\mu}_L$ will denote the $\lvert\tau_B\rvert$- and $\lvert\tau_L\rvert$-vectors that involve these coefficients, respectively, $\boldsymbol{\mu}_B = (\mu_t)_{t\in\tau_B}$ and $\boldsymbol{\mu}_L = (\mu_t)_{t\in\tau_L}$. Let $\boldsymbol{\mu}$ denote the $(\lvert\tau_B\rvert + \lvert\tau_L\rvert)$-vector $\boldsymbol{\mu} = (\boldsymbol{\mu}_B, \boldsymbol{\mu}_L)$. Both notations will be used interchangeably when needed.
Probabilities	
$p_{it}(\boldsymbol{a}_{\cdot t}, \mu_t)$	probability of individual i going down the left branch at branch node t. Its expression is $p_{it}(\boldsymbol{a}_{\cdot t}, \mu_t) = F\left(\frac{1}{p}\boldsymbol{a}_{\cdot t}^\top\boldsymbol{x}_i - \mu_t\right),\ i \in \mathcal{I},\ t \in \tau_B$,
$P_{it}(\boldsymbol{a}_B, \boldsymbol{\mu}_B)$	probability of individual i falling into leaf node t. Its expression is $P_{it}(\boldsymbol{a}_B, \boldsymbol{\mu}_B) = \prod_{t_l\in N_L(t)} p_{it_l}\left(\boldsymbol{a}_{\cdot t_l}, \mu_{t_l}\right) \prod_{t_r\in N_R(t)} (1 - p_{it_r}(\boldsymbol{a}_{\cdot t_r}, \mu_{t_r})), i \in \mathcal{I},\ t \in \tau_L$.

(continued on next page)

Predictions	
$\varphi_{it}(\boldsymbol{a}_{\cdot t}, \mu_t)$	linear prediction of individual i at leaf node t. Its expression is $\varphi_{it}(\boldsymbol{a}_{\cdot t}, \mu_t) = \boldsymbol{a}_{\cdot t}^{\top}\boldsymbol{x}_i - \mu_t,\ i \in \mathcal{I},\ t \in \tau_L$,
$\varphi_i(\boldsymbol{a}, \boldsymbol{\mu})$	final prediction of individual i. Its expression is $\varphi_i(\boldsymbol{a}, \boldsymbol{\mu}) = \sum_{t\in\tau_L} P_{it}(\boldsymbol{a}_B, \boldsymbol{\mu}_B)\varphi_{it}(\boldsymbol{a}_{\cdot t}, \mu_t),\ i \in \mathcal{I}$. In other words, for an individual i, its prediction is a weighted average of the predictions φ_{it} along the different leaf nodes, where the weights in such average depend on the individual i.

2.2. The formulation

With these parameters and decision variables, the S-ORRT reads as the following unconstrained NLCO problem:

$$\min_{\boldsymbol{a}, \boldsymbol{\mu}} \left\{ \text{MSE}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{I}) \ + \lambda^L \sum_{j=1}^{p} \left\| \boldsymbol{a}_{j\cdot} \right\|_1 + \lambda^G \sum_{j=1}^{p} \left\| \boldsymbol{a}_{j\cdot} \right\|_\infty \right\}, \tag{1}$$

where

$$\text{MSE}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{I}) = \frac{1}{|\mathcal{I}|} \sum_{i\in\mathcal{I}} (\varphi_i(\boldsymbol{a}, \boldsymbol{\mu}) - y_i)^2.$$

The first term, prediction accuracy, is equal to the mean squared error over the training sample between the actual response values and the predictions returned by S-ORRT. The second term controls local sparsity, since it penalizes the ℓ_1-norm of the coefficients of the predictor variables used in the cuts along the tree. The third term addresses global sparsity, which is modeled by the inclusion of a penalization term that controls whether a given predictor variable is ever used across the whole tree. Recall that each predictor variable appears at both branch (in the oblique cuts) and leaf (in the linear predictions) nodes. Then, the ℓ_∞-norm is used as a group penalty function, by forcing all the coefficients linked to the same predictor variable to be shrunk simultaneously along all branch and leaf nodes.

Since there are no decision variables directly relating to the number of individuals N, Problem (1) speaks favorably toward the scalability of S-ORRT with respect to the size of the training sample. Hence, although the evaluation of the first term in the objective function becomes more time demanding with larger N, the number of decision variables of the problem to be solved remains the same. This makes our approach scalable with respect to N, as illustrated in Section 4.4.

Once the tree model is built, the prediction of future data is done as follows. Let $(\boldsymbol{a}^*, \boldsymbol{\mu}^*)$ be the optimal solution to Problem (1). The expected outcome of individual $i \in \mathcal{I}$ is $\varphi_i(\boldsymbol{a}^*, \boldsymbol{\mu}^*)$. For an incoming individual with predictor vector $\mathbf{x}$, the expected outcome returned by the randomized tree is equal to

$$\mathbf{x} \rightarrow \Pi(\mathbf{x}) := \varphi_{\mathbf{x}}(\boldsymbol{a}^*, \boldsymbol{\mu}^*), \tag{2}$$

where $\varphi_{\mathbf{x}}$ is defined similarly to φ_i with $\mathbf{x}$ replacing $\mathbf{x}_i$. Note that $\Pi(\cdot)$ is smooth in the continuous predictor variables, since the CDF F is assumed to be a smooth function. This means that even small changes in these variables will produce changes in $\Pi(\cdot)$. This is not the case for deterministic tree models such as CART and RF, where there are no changes at all in the expected outcome when there are small changes in the continuous predictor variables. This inherent property of our approach allows us to perform local explainability, as will be seen in Section 2.4.

2.3. A smooth reformulation

Problem (1) is non-smooth due to the ℓ_1 and ℓ_∞ norms appearing in the objective function. Recall that F is assumed to be continuously differentiable, therefore MSE inherits smoothness. By rewriting both regularization terms using new decision variables, we can formulate S-ORRT as a smooth problem, thus solvable with standard continuous optimization solvers, as done in our computational section.

Regarding the first regularization term of Problem (1), decision variables $\mathbf{a}$ are split into their positive and negative coun-

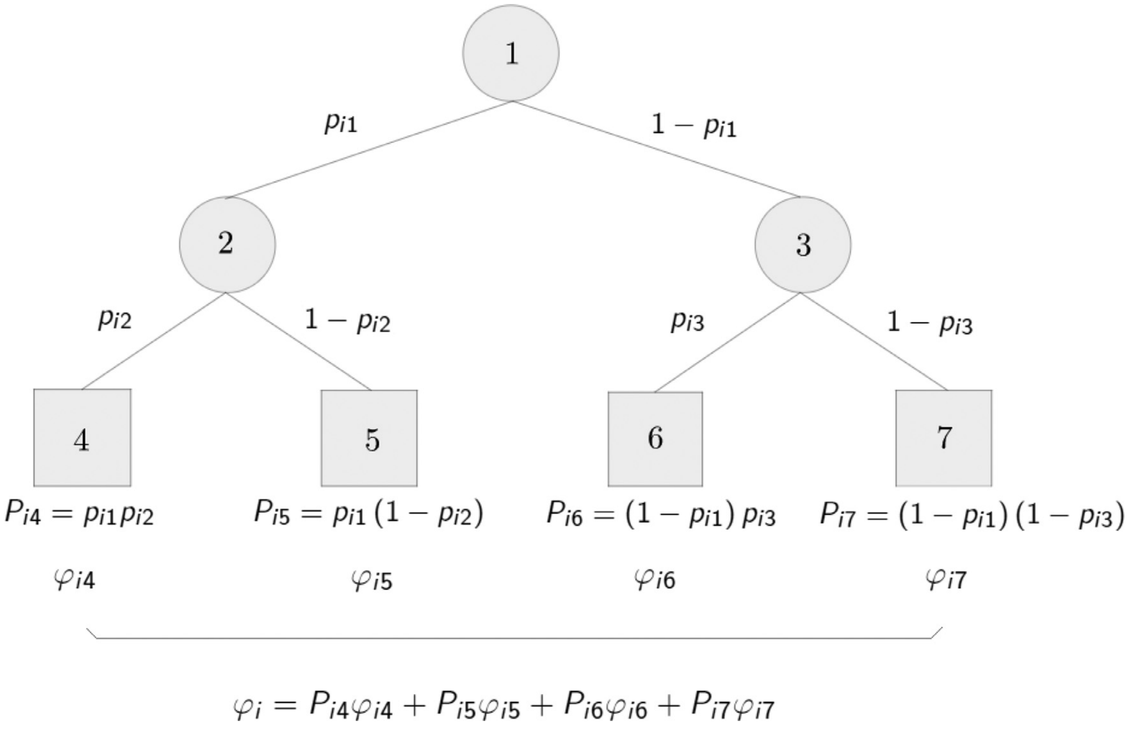
1
p_{i1}
$1 - p_{i1}$
2
3
p_{i2}
$1 - p_{i2}$
p_{i3}
$1 - p_{i3}$
4
5
6
7
$P_{i4} = p_{i1}p_{i2}$
$P_{i5} = p_{i1}(1 - p_{i2})$
$P_{i6} = (1 - p_{i1})p_{i3}$
$P_{i7} = (1 - p_{i1})(1 - p_{i3})$
φ_{i4}
φ_{i5}
φ_{i6}
φ_{i7}
$\varphi_i = P_{i4}\varphi_{i4} + P_{i5}\varphi_{i5} + P_{i6}\varphi_{i6} + P_{i7}\varphi_{i7}$

Fig. 1. Sparse Optimal Randomized Regression Tree of depth $D = 2$.

terparts, $\boldsymbol{a}^+ = \left(a^+_{jt}\right)_{j=1,\ldots,p,\ t\in\tau_B\cup\tau_L}$ and $\boldsymbol{a}^- = \left(a^-_{jt}\right)_{j=1,\ldots,p,\ t\in\tau_B\cup\tau_L}$, respectively, such that $a_{jt} = a^+_{jt} - a^-_{jt}$, $\left|a_{jt}\right| = a^+_{jt} + a^-_{jt}$ and $a^+_{jt}, a^-_{jt} \geq 0$, thus having

$$\left\|\boldsymbol{a}_{j\cdot}\right\|_1 = \sum_{t\in\tau_B\cup\tau_L} \left|a_{jt}\right| = \sum_{t\in\tau_B\cup\tau_L} \left(a^+_{jt} + a^-_{jt}\right),\ \ j = 1, \ldots, p.$$

New decision variables $\boldsymbol{\beta} = \left(\beta_j\right)_{j=1,\ldots,p}$ are used to model the second regularization term of Problem (1):

$$\left\|\boldsymbol{a}_{j\cdot}\right\|_\infty = \max_{t\in\tau_B\cup\tau_L} \left|a_{jt}\right| = \beta_j,\ \ j = 1, \ldots, p,$$

where $\beta_j \geq 0$. We also need to impose $\beta_j \geq \left|a_{jt}\right| = a^+_{jt} + a^-_{jt},\ j = 1, \ldots, p, t \in \tau_B \cup \tau_L$. Hence, we have that Problem (1) is equivalent to the following smooth reformulation:

$$\min_{\boldsymbol{a}^+,\boldsymbol{a}^-,\,\boldsymbol{\mu},\,\boldsymbol{\beta}} \quad \mathrm{MSE}\big(\boldsymbol{a}^+ - \boldsymbol{a}^-, \boldsymbol{\mu}; \mathcal{I}\big) + \lambda^L \sum_{j=1}^{p} \sum_{t\in\tau_B\cup\tau_L} \left(a^+_{jt} + a^-_{jt}\right) + \lambda^G \sum_{j=1}^{p} \beta_j \tag{3}$$

$$\text{s.t.} \quad \beta_j \geq a^+_{jt} + a^-_{jt},\ \ j = 1, \ldots, p,\ \ t \in \tau_B \cup \tau_L, \tag{4}$$

$$a^+_{jt},\ a^-_{jt},\ \beta_j \geq 0,\ \ j = 1, \ldots, p,\ \ t \in \tau_B \cup \tau_L. \tag{5}$$

2.4. Desirable properties

As we show in this section, our approach can easily accommodate important desirable properties in the regression task, such as cost-sensitivity and fairness, as well as local explainability.

Cost-sensitivity

As a regression method, S-ORRT seeks a rule yielding a good overall prediction accuracy, although, at times, there are groups of individuals in which predicion errors are more critical. It is then more adequate not only to focus on the overall prediction accuracy, but also ensuring a certain level of performance in those groups. S-ORRT is flexible enough to allow incorporating constraints on expected performance (Blanquero, Carrizosa, Ramírez-Cobo, & Sillero-Denamiel, 2021b) over critical groups. Let $\mathcal{J}_1, \ldots, \mathcal{J}_r$ be different samples, possibly subsamples of $\mathcal{I}$. Given a threshold value ρ_j for the desired performance on sample $\mathcal{J}_j$, one can simply add the following constraints to Problem (1):

$$\mathrm{MSE}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{J}_j) \leq \rho_j, \quad j = 1, \ldots, r.$$

Fairness

The increase of automatization in decision-making have evinced the bias present on historical data, leading to models that may discriminate groups sharing sensitive features such as gender or race. In this line, we seek for a model that avoids such discrimination and is fair to a sentive group. Let $\mathcal{S} \subset \mathcal{I}$ be a group of individuals to be protected against discrimination by Problem (1). There are different ways to handle fairness. For instance, we may impose that the prediction errors for individuals in $\mathcal{S}$ does not differ much from the prediction errors in the whole training sample $\mathcal{I}$. This can be modeled through the following constraint

$$|\mathrm{MSE}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{S}) - \mathrm{MSE}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{I})| \leq C,$$

for $C \geq 0$ sufficiently small. Alternatively, we may impose that the average prediction for individuals in $\mathcal{S}$ does not differ much from the average in the whole training sample $\mathcal{I}$, i.e.,

$$|\bar{\varphi}(\boldsymbol{a}, \boldsymbol{\mu}; S) - \bar{\varphi}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{I})| \leq C, \tag{6}$$

where $\bar{\varphi}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{J}) = \frac{1}{|\mathcal{J}|} \sum_{i\in\mathcal{J}} \varphi_i(\boldsymbol{a}, \boldsymbol{\mu})$ and $C \geq 0$ sufficiently small. Fairness as in Eq. (6) is illustrated for the `Boston Housing` data set (Harrison & Rubinfeld, 1978). See Table 2 for a description of the response and predictor variables. Suppose that our sensitive group $\mathcal{S}$ is composed by individuals above the third quartile of predictor variable B, that is, those census tracts where there is a high proportion of black population. The S-ORRT without fairness constraints, and $\lambda^L = \lambda^G = 0$, yields a mean squared error of 9.6462, with an average prediction on housing values over $\mathcal{I}$ equal to 22.5333. A lower average value is obtained over $\mathcal{S}$, 21.3263, producing an absolute difference of $C_0 = 1.2070$. See the first row in Table 1. The next rows represent the results when fairness constraints over $\mathcal{S}$ are added to the model for several values of the threshold $C = \tau \cdot C_0$, with τ varying in $\{0.75, 0.5, 0.25, 0\}$. As shown, one can obtain an S-ORRT which is fair to our sensitive group $\mathcal{S}$, since $\bar{\varphi}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{I}) = \bar{\varphi}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{S})$, at the expense of slightly harming prediction accuracy.

Local explainability

The goal of local explainability is to identify the predictor variables that have the largest impact on the individual predictions, found in Eq. (2). As opposed to post-hoc approaches, we can directly derive local explanations on the continuous predictor variables thanks to the smoothness of Π. For simplicity, we consider a problem where all predictor variables are continuous. For an individual with predictor variables $\boldsymbol{x}^0$, we analyze how sensitive Π is to an infinitesimal change $\boldsymbol{\Delta} \in \mathbb{R}^p$, i.e., how large is the difference $\Pi(\boldsymbol{x}^0 + \boldsymbol{\Delta}) - \Pi(\boldsymbol{x}^0)$. By linearizing Π close to $\boldsymbol{x}^0$, we have

$$\Pi(\boldsymbol{x}^0 + \boldsymbol{\Delta}) \approx \Pi(\boldsymbol{x}^0) + \sum_{j=1}^{p} \frac{\partial \Pi}{\partial x_j}(\boldsymbol{x}^0) \cdot \Delta_j.$$

Thus, the vector of partial derivatives

$$\left(\frac{\partial \Pi}{\partial x_j}(\boldsymbol{x}^0)\right)_{j=1,\ldots,p} \tag{7}$$

gives full information on the sensitivity of the outcomes Π around $\boldsymbol{x}^0$. A positive value of coordinate j of the vector of partial derivatives means a direct relationship between predictor variable j and prediction of the response variable of individual $\boldsymbol{x}^0$; and an inverse relationship, otherwise. As opposed to linear regression, where there is one single coefficient per predictor variable that indicates its impact in prediction for any individual equally, here we have different impacts of each predictor variable tailored to each particular individual.

Local explainability is illustrated below for the `Boston Housing` data set in Table 2.

An S-ORRT with $\lambda^L = 0$ and $\lambda^G = \frac{2^2}{13}$ was built on this data set, obtaining a mean squared error and an R-squared equal to 15.5654 and 0.8156, respectively. Figure 2 depicts the local explanations for all individuals in the dataset by means of parallel coordinates. Each predictor variable is represented by a vertical parallel axis. Each individual is represented by a series of lines connected across all the axes. The position each individual takes on each axis reflects the impact the corresponding predictor variable has on its prediction, that is, each of the coordinates of vector (7). The color that represents each individual in the parallel coordinates goes from light pink to purple depending on the reliability on prediction, measured as the ratio between the individual squared error and the mean squared error. Thus, purple refers to the best reliable predictions according to the model. All predictor variables were normalized before training the model to the 0–1 interval, in such a way that a fair comparative analysis between them could be performed. A larger absolute value on the axis represents a larger impact caused by the corresponding predictor variable on the prediction. Since `CRIM` gauges the threat to well-being that households perceive, it has a negative effect on housing values. A similar pattern is observed for `NOX`, `DIS`, `TAX`, `PTRATIO`, as well as for `LSTAT`, which means that an area with a high amount of lower

Table 2
Information about the Boston Housing data set, which consists of a collection of 506 observations about housing values for census tracts of the Boston metropolitan area.

Variable	Name	Description
Predictor	CRIM	crime rate by town
	ZN	proportion of residential land zoned for lots greater than 25,000 squared feet
	INDUS	proportion of nonretail business acres per town
	CHAS	1 if tract bounds river; 0 otherwise
	NOX	nitrogen oxide concentration in parts per hundred million
	RM	average number of rooms in owner units
	AGE	proportion of owner units built prior to 1940
	DIS	weighted distances to five employment centers in the Boston region
	RAD	index of accessibility to radial highways
	TAX	full value property tax rate per ten thousands of dollars
	PTRATIO	pupil-teacher ratio by town school district
	B	black proportion of population
	LSTAT	proportion of population that is lower status
Response	MEDV	median value of owner-occupied homes in thousands of dollars

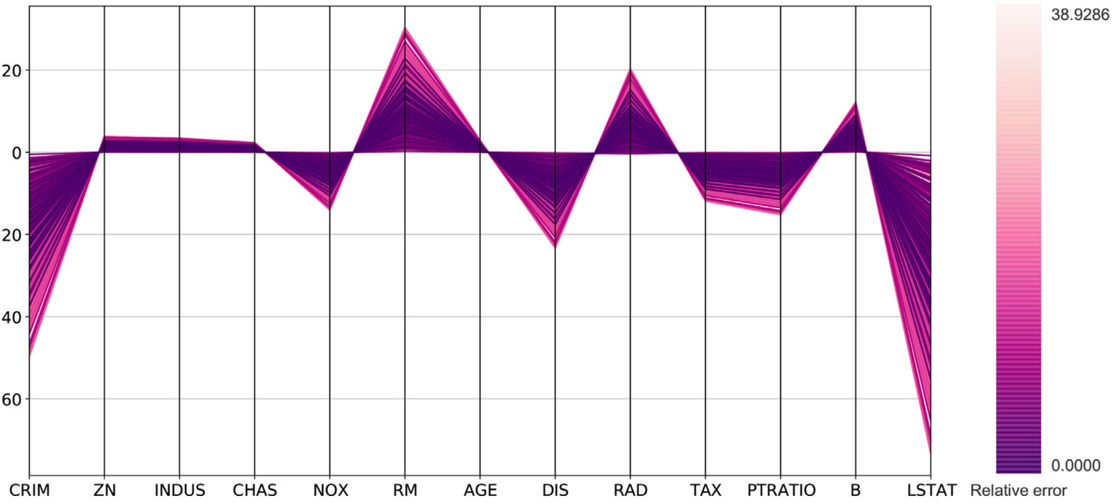

Fig. 2. Local explainability for Boston Housing data set derived from the S-ORRT with $\lambda^L = 0$ and $\lambda^G = \frac{2^2}{13}$ and a mean squared error and an R-squared of 15.5654 and 0.8156, respectively.

Table 1
Results of S-ORRT without and with fairness constraints on $\mathcal{S}$ in the Boston Housing data set, where $C_0 = 1.2070$.

τ	$C = \tau \cdot C_0$	$\text{MSE}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{I})$	$\bar{\varphi}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{I})$	$\bar{\varphi}(\boldsymbol{a}, \boldsymbol{\mu}; \mathcal{S})$
-	-	9.6462	22.5333	21.3263
0.75	0.9053	9.7586	22.5327	21.6275
0.5	0.6035	10.0282	22.5334	21.9298
0.25	0.3018	10.5051	22.5330	22.1312
0	0	11.2401	22.5332	22.5332

status population would have less valuable households. Other predictor variables have a positive effect on housing values. For instance, RM, which represents spaciousness and it can be observed that is directly related to a higher housing value.

3. Theoretical properties

In this section, some theoretical properties enjoyed by S-ORRT, as formulated in Problem (1), are analyzed. In particular, we pay attention to the most sparse tree, obtained when the optimal solution of S-ORRT includes $\boldsymbol{a}^* = \boldsymbol{0}$, and thus none predictor variable is used in the predictions. This is attained when the sparsity regularization parameters, λ^L and λ^G, are taken large enough, and the first term related to the prediction accuracy of the regressor becomes negligible. In the following, we study the optimal prediction returned by S-ORRT with $\boldsymbol{a}^* = \boldsymbol{0}$, and derive upper bounds for λ^L and λ^G in the sense that above them the most sparse tree (with $\boldsymbol{a}^* = \boldsymbol{0}$) is a stationary point of the S-ORRT, that is, there exists $(\boldsymbol{a}^* = \boldsymbol{0}, \boldsymbol{\mu}^*)$ such that the necessary optimality condition with respect to $\boldsymbol{a}$ is satisfied. In Section 4, we illustrate when these upper bounds are already reached, by showing that above certain values of λ^L and λ^G, the highest levels of local and global sparsity, respectively, are achieved. See in Fig. 3 that for $\left(\lambda^L, \lambda^G\right) = \left(\frac{2^2}{120}, \frac{2^2}{40}\right)$, the most sparse S-ORRT is already obtained, while not producing the best performance in terms of prediction accuracy.

First, observe that, for any $\boldsymbol{a}$ and $\boldsymbol{\mu}_B$ fixed, Problem (1) can be easily reformulated as a linear regression problem. Indeed, we have that the final prediction of each individual is

$$\varphi_i(\boldsymbol{a}, \boldsymbol{\mu}) = \sum_{t \in \tau_L} P_{it}(\boldsymbol{a}_B, \boldsymbol{\mu}_B)\left(\boldsymbol{a}_{\cdot t}^\top \boldsymbol{x}_i - \mu_t\right), \ i \in \mathcal{I},$$

and thus, defining

$$\eta_i(\boldsymbol{a}, \boldsymbol{\mu}_B) = \sum_{t \in \tau_L} P_{it}(\boldsymbol{a}_B, \boldsymbol{\mu}_B)\left(\boldsymbol{a}_{\cdot t}^\top \boldsymbol{x}_i - y_i\right), \ i \in \mathcal{I},$$

the MSE term in Problem (1) can be rewritten as

$$\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \left(\eta_i(\boldsymbol{a}, \boldsymbol{\mu}_B) - \sum_{t \in \tau_L} P_{it}(\boldsymbol{a}_B, \boldsymbol{\mu}_B)\mu_t \right)^2,$$

or, in matrix form,

$$\frac{1}{|\mathcal{I}|} \|\boldsymbol{\eta}(\boldsymbol{a}, \boldsymbol{\mu}_B) - \boldsymbol{P}(\boldsymbol{a}_B, \boldsymbol{\mu}_B)\boldsymbol{\mu}_L\|^2,$$

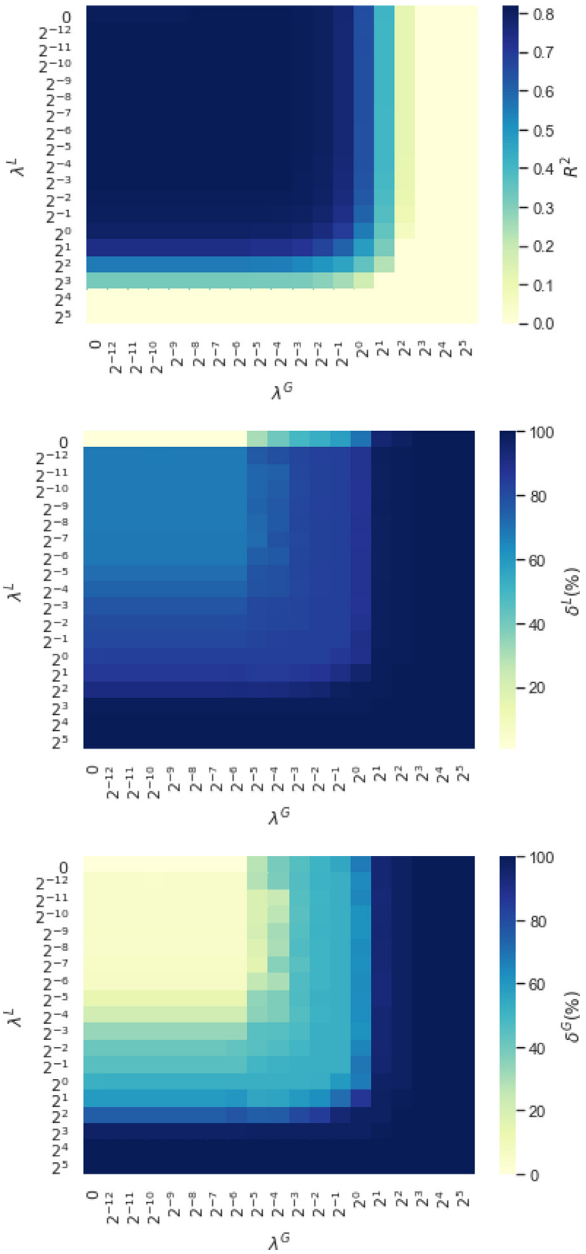

Fig. 3. Heatmaps representation, for Ailerons data set, of the average R-squared obtained, R^2, the average percentage of predictor variables not used per node, δ^L, and the average percentage of predictor variables not used per tree, δ^G, respectively, as a function of the grid of the sparsity regularization parameters, λ^L and λ^G, considered in the S-ORRT construction.

where

$$\boldsymbol{\eta}(\boldsymbol{a}, \boldsymbol{\mu}_B) = (\eta_i(\boldsymbol{a}, \boldsymbol{\mu}_B))_{i\in\mathcal{I}}$$

and

$$\boldsymbol{P}(\boldsymbol{a}_B, \boldsymbol{\mu}_B) = \begin{bmatrix} P_{it}(\boldsymbol{a}_B, \boldsymbol{\mu}_B) \end{bmatrix}_{i\in\mathcal{I},\, t\in\tau_L}.$$

Then, minimizing MSE for $\boldsymbol{a}, \boldsymbol{\mu}_B$ fixed amounts to finding the Ordinary Least Squares solution with design matrix $\boldsymbol{P}(\boldsymbol{a}_B, \boldsymbol{\mu}_B)$ and response vector $\boldsymbol{\eta}(\boldsymbol{a}, \boldsymbol{\mu}_B)$. With this, the following is shown:

Proposition 1. *For $\left(\boldsymbol{a}^*, \boldsymbol{\mu}_B^*\right)$ fixed, $\boldsymbol{\mu}_L^*$ minimizes $\mathrm{MSE}\left(\boldsymbol{a}^*, \left(\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L\right)\right)$ if, and only if,*

$$\boldsymbol{P}^\top(\boldsymbol{a}_B^*, \boldsymbol{\mu}_B^*)\boldsymbol{\eta}(\boldsymbol{a}^*, \boldsymbol{\mu}_B^*) = \boldsymbol{P}^\top(\boldsymbol{a}_B^*, \boldsymbol{\mu}_B^*)\boldsymbol{P}(\boldsymbol{a}_B^*, \boldsymbol{\mu}_B^*)\boldsymbol{\mu}_L^*.$$

In particular, for the most sparse solution $\boldsymbol{a}^* = \boldsymbol{0}$, we have the following corollary.

Table 3
Information about the real-world data sets considered.

Data set	Abbreviation	N	p
Boston-housing	BH	506	13
Red-wine	RW	1599	11
White-wine	WW	4898	11
Parkinson-motor	PM	5874	16
Parkinson-total	PT	5874	16
Ailerons	A	7153	40
Cpu-act	CA	8192	21
Cart-artificial	CAr	40,768	10
Friedman-artificial	FA	40,768	10

Corollary 1. *For any $\boldsymbol{\mu}_B^*$, the vector $\boldsymbol{\mu}_L^* = \begin{pmatrix} -\bar{y}, & \cdots, & -\bar{y} \end{pmatrix}^\top$, with $\bar{y} = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} y_i$, minimizes $\mathrm{MSE}\left(\boldsymbol{a}^* = \boldsymbol{0}, \left(\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L\right); \mathcal{I}\right)$, and then the prediction is $\varphi_i\left(\boldsymbol{a}^* = \boldsymbol{0}, \left(\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*\right)\right) = \bar{y}$ for all $i \in \mathcal{I}$.*

Proof. See Appendix. □

As stated, when λ^L and λ^G are taken large enough in Problem (1), the most sparse possible tree (with $\boldsymbol{a}^* = \boldsymbol{0}$) is obtained though possibly not yielding the best prediction accuracy, since none of the predictor variables is used to fit the model. As observed in Fig. 3, it turns out that the solution $\boldsymbol{a}^* = \boldsymbol{0}$ is not only the limit case when λ^L and λ^G tend to infinity, but it is actually obtained already for finite values of them. This is shown in the following.

Proposition 2. *Let $\boldsymbol{a}^* = \boldsymbol{0}$, $\boldsymbol{\mu}_B^* \in \mathbb{R}^{|\tau_B|}$, and $\boldsymbol{\mu}_L^* = \begin{pmatrix} -\bar{y}, & \cdots, & -\bar{y} \end{pmatrix}^\top$. Let $\sigma \in [0, 1]$,*

$$\lambda^l = (1-\sigma)\max_{j=1,\ldots,p}\left\|\nabla_{\boldsymbol{a}_{j\cdot}}\mathrm{MSE}(\boldsymbol{0}, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I})\right\|_\infty \text{ and}$$

$$\lambda^g = \quad \sigma \max_{j=1,\ldots,p}\left\|\nabla_{\boldsymbol{a}_{j\cdot}}\mathrm{MSE}(\boldsymbol{0}, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I})\right\|_1.$$

Then, for any pair $\left(\lambda^L, \lambda^G\right)$ such that $\lambda^L \geq \lambda^l$ and $\lambda^G \geq \lambda^g$, $\left(\boldsymbol{a}^*, \left(\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*\right)\right)$ is a stationary point of Problem (1).

Proof. See Appendix. □

4. Computational experiments

The aim of this section is to illustrate the performance of our sparse optimal randomized regression trees (S-ORRT) using both real-world and synthetic data sets. Section 4.1 gives details on the procedure followed to test our approach in the real-world data sets. In Section 4.2 we discuss the prediction accuracy of S-ORRT, against several benchmark regression methods. In Section 4.3 we illustrate our ability to trade in some of the prediction accuracy of S-ORRT for a gain in local and global sparsity. Finally, in Section 4.4 we illustrate the scalability of S-ORRT in terms of the number of individuals in the training sample, using a synthetic data set.

4.1. Setup

A collection of well-known real-world data sets from the UCI Machine Learning Repository (Lichman, 2013) has been chosen. Table 3 lists their names, the abbreviations used throughout this section to refer to them, together with their number of observations and predictor variables.

Each data set has been randomly split into two subsets: the training subset (75%) and the test subset (25%). The corresponding tree model is built on the training subset and, then, three performance criteria, namely prediction accuracy, local and global sparsity, are assessed. The prediction accuracy is evaluated by the out-

Table 4
Comparison between S-ORRT with $\lambda^L = \lambda^G = 0$, CART, OLS, LASSO, ORT-H LS and RF in terms of out-of-sample R-squared, R^2, on real-world data sets in Table 3.

Data set	Out-of-sample average R^2							
	CART	OLS	LASSO	ORT-H LS	RF	S-ORRT $D=1$	S-ORRT $D=2$	S-ORRT $D=3$
BH	0.7416(5)	0.7391(7)	0.7401(6)	0.8040(2)	**0.8759**(1)	0.5987(8)	0.7931(3)	0.7785(4)
RW	0.3055(7)	0.3619(3)	0.3605(5)	0.3040(8)	**0.4874**(1)	0.3482(6)	0.3730(2)	0.3613(4)
WW	0.2539(8)	0.2714(6)	0.2699(7)	0.3490(2)	**0.5196**(1)	0.3121(5)	0.3291(4)	0.3337(3)
PM	0.1020(6)	0.0878(8)	0.0900(7)	0.2810(2)	**0.3426**(1)	0.1878(5)	0.2121(4)	0.2400(3)
PT	0.1294(6)	0.0849(8)	0.0863(7)	0.3160(2)	**0.3545**(1)	0.1724(5)	0.1965(4)	0.2445(3)
A	0.6466(8)	0.8167(7)	0.8173(6)	**0.8360**(1)	0.8211(3)	0.8207(5)	0.8288(2)	0.8211(3)
CA	0.9324(5)	0.7272(8)	0.7273(7)	**0.9840**(1)	0.9829(2)	0.8282(6)	0.9535(4)	0.9540(3)
CAr	0.8771(6)	0.7045(7)	0.7045(7)	**0.9480**(1)	0.9425(5)	**0.9480**(1)	**0.9480**(1)	**0.9480**(1)
FA	0.6058(8)	0.7222(7)	0.7223(6)	**0.9560**(1)	0.9245(4)	0.8493(5)	0.9501(3)	0.9505(2)
Average	0.5105(6.5)	0.5017(6.7)	0.5020(6.4)	0.6420(2.2)	**0.6946**(2.1)	0.5628(5.1)	0.6205(3.0)	0.6257(2.8)

of-sample R-squared (R^2) in the test subset:

$$R^2 = 1 - \frac{\mathrm{MSE}_{\mathrm{test}}}{\mathrm{V}_{\mathrm{test}}},$$

where $\mathrm{MSE}_{\mathrm{test}}$ is the mean squared error obtained by the regression method in the test subset and $\mathrm{V}_{\mathrm{test}}$ is the variance of the actual response vector in the test subset too. The higher the R^2, the better the model in terms of prediction accuracy.

The control of local and global sparsity is one of the key features of S-ORRT, as has been pointed out previously. Local sparsity, δ^L, is measured as the average percentage of predictor variables not used per node:

$$\delta^L = \frac{1}{|\tau_B| + |\tau_L|} \sum_{t \in \tau_B \cup \tau_L} \frac{\left|\left\{a_{jt} = 0,\ \ j = 1, \ldots, p\right\}\right|}{p} \times 100.$$

Global sparsity, δ^G, is measured as the percentage of predictor variables not used at any of the nodes, i.e., across the whole tree:

$$\delta^G = \frac{\left|\left\{\boldsymbol{a}_{j\cdot} = \boldsymbol{0},\ \ j = 1, \ldots, p\right\}\right|}{p} \times 100.$$

The higher δ^L and δ^G, the better the model in terms of local and global sparsity, respectively.

The training/testing procedure has been repeated ten times. The results shown in Table 4 and Fig. 3 represent the average of such ten runs for the above-mentioned performance criteria.

The logistic CDF has been chosen for our experiments:

$$F(\cdot) = \frac{1}{1 + \exp\left(-(\cdot)\gamma\right)},$$

with a large value of γ, namely, $\gamma = 512$. We will illustrate that this small level of randomization is enough for obtaining good results.

The S-ORRT smooth formulation (3)–(5) has been implemented using the `scipy.optimize` package (Jones, Oliphant, Peterson et al., 2001) in Python 3.7 (Python Core Team, 2015). As a solver, we have used the SLSQP method (Kraft, 1988) that allows one to use gradient information. The predictor variables have been previously normalized to the $[0, 1]$ interval, and the decision variables $\boldsymbol{a}_B$ and $\boldsymbol{\mu}_B$ have been restricted to the $[-1, 1]$ interval. Our experiments have been conducted on a PC, with an Intel®Core$^{\mathrm{TM}}$ i7-9700 CPU 3.00 GHz processor (8 CPUs) and 64 GB RAM. The operating system is 64 bits.

4.2. Comparison of prediction performance

In this section we focus on illustrating the prediction accuracy of all the methods tested on the real-world data sets. S-ORRT at depths $D = 1,\ 2$ and 3 with $\lambda^L = \lambda^G = 0$ is compared against three types of benchmark regression methods. The first type corresponds to standard regression methods, such as CART, the classic approach to build decision trees, with no restrictions on depth, and OLS. The second type is the leader regression method in terms of sparsity, LASSO. Finally, in the third type we have two sophisticated tree-based regression methods competitive in terms of prediction accuracy, such as ORT-H LS in Dunn (2018), a Mathematical Programming based approach that employs a local-search heuristic for building oblique trees with linear predictions at maximum depth $D = 10$; and Random Forest (RF), an ensemble of CARTs using a boostrap aggregating scheme. Table 4 presents the average out-of-sample prediction accuracy R^2, while in parenthesis we show how the method ranks in terms of its prediction accuracy. For a given data set, a rank of “1” indicates that the method is the best in terms of out-of-sample R^2 while a rank of “8” indicates that the method performed the worst. The average R^2 and rank of each method across all data sets are found at the bottom of the table.

For S-ORRT, we have followed a multistart approach, where the process is repeated 1000 times starting from different random initial solutions. For a given initial solution, the computing time taken by the S-ORRT typically ranges from 0.01 s (in BH for $D = 1$) to 2.08 s (in A and FA for $D = 3$). The default parameter setting in `rpart` (Therneau, Atkinson, & Ripley, 2015), `glmnet` (Friedman, Hastie, & Tibshirani, 2010) and `randomForest` (Liaw & Wiener, 2002) R packages have been used for running CART, OLS and LASSO, and RF, respectively. For ORT-H LS, the results are taken from (Dunn, 2018), since open-source implementations were not available.

We start discussing the results for our S-ORRT with depth $D = 3$. S-ORRT outperforms CART, OLS and LASSO, yielding increases in the R^2 up to 34 percentage points (p.p.) with respect to CART, and up to 24 p.p. with respect to OLS and LASSO, both with comparable performance. Regarding ORT-H LS, S-ORRT presents an average prediction accuracy 2 p.p. lower, however S-ORRT manages to be comparable in CAr and outperform in RW by 6 p.p. Finally, although RF reports the best overall performance across all the methods, S-ORRT is comparable to RF in A and CAr, while S-ORRT has the best prediction accuracy in FA.

With depth $D = 2$, the conclusions for S-ORRT are similar to those obtained using depth $D = 3$. With depth $D = 1$, S-ORRT still manages to be powerful in some data sets, despite the low complexity of the model. S-ORRT outperforms CART, OLS and LASSO in six of the data sets considered, all except for BH, RW and CA. ORT-H LS generally outperforms S-ORRT at depth $D = 1$, with the exception of CAr, where S-ORRT is comparable, and RW, where S-ORRT is superior in 4 p.p. With respect to RF, S-ORRT is outperformed in general, but has a comparable prediction accuracy in A and CAr.

In summary, these numerical results illustrate that, in terms of prediction accuracy, S-ORRT with $D = 2, 3$ outperforms the

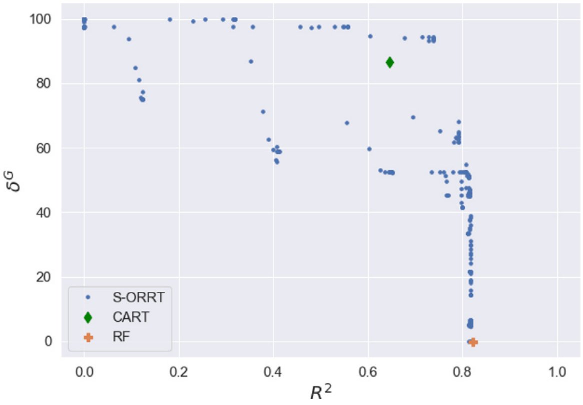

Fig. 4. Scatterplot representation, for Ailerons data set, of the average R-squared obtained, R^2, and the average percentage of predictor variables not used per tree, δ^G. Blue points refer to the solution of every pair of the sparsity regularization parameters $\left(\lambda^L, \lambda^G\right)$ considered in the S-ORRT construction; the green diamond, to CART solution; and the orange cross, to RF solution. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

standard benchmark regression methods (CART and OLS) and the benchmark regression method in sparsity (LASSO). Regarding more sophisticated tree-based approaches, ORT-H LS and RF show slightly better prediction accuracies, although S-ORRT is competitive in some data sets. Unlike CART, ORT-H LS and RF, our approach has a direct control on global desirable properties such as sparsity, cost-sensitivity and fairness.

4.3. Prediction accuracy and sparsity tradeoff

The aim of this section is to illustrate that, in contrast to sophisticated tree-based regression methods that rely on greedy or local-search approaches, such as RF and ORT-H LS, our S-ORRT is able to trade in some of its prediction accuracy for a gain in local and global sparsity. For the sake of conciseness, we illustrate this in the Ailerons data set. We have solved Problem (3)–(5) with depth $D = 1$ for the sparsity parameters λ^L and λ^G in a grid. We have taken the grid $\{0\} \cup \{2^r, \ -12 \le r \le 5, \ r \in \mathbb{Z}\}$, normalized by the number of predictor variables, and in the case of λ^L by the number of nodes too. We start solving the optimization problem with $\left(\lambda^L, \lambda^G\right) = (0, 0)$. We continue with $\lambda^L = 0$ but for larger values of λ^G. Once all $\left(0, \lambda^G\right)$ are executed, we start the process all over again with the next value of λ^L in the grid. The solutions found to Problem (3)–(5) for fixed $\left(\lambda^L, \lambda^G\right)$, are given as initial solutions to the next problem to be solved in the grid.

Figure 3 illustrates these results by means of three heatmaps: one for the prediction accuracy, R^2, another one for the local sparsity, δ^L, and the final one for the global sparsity, δ^G. The color bar of each heatmap goes from light green to dark blue, the latter indicating the best (maximum) R^2, δ^L or δ^G achieved, respectively. By definition, the sparsest tree is obtained for large of values of λ^L, λ^G. We can observe that the best rates of prediction accuracy are not only achieved for $\left(\lambda^L, \lambda^G\right) = (0, 0)$. Clearly, the R^2 remains almost constant for pair of values $\left(\lambda^L, \lambda^G\right)$ that verify $\lambda^L \le \bar{\lambda}^L$ and $\lambda^G \le \bar{\lambda}^G$ where $\left(\bar{\lambda}^L, \bar{\lambda}^G\right) = \left(\frac{2^{-2}}{120}, \frac{2^{-4}}{40}\right)$. In this range, where we have the best prediction accuracy, we can dramatically enhance both the local and the global sparsity. Indeed, the local sparsity improves from 1% to 84% and the global sparsity from 0% to 52%. For larger values of λ^L and λ^G, our S-ORRT keeps improving sparsity but, in this case, at the cost of diminishing R^2.

Figure 4 reflects, against CART and RF, our ability to trade off prediction accuracy and global sparsity in Ailerons data set. The value of both performance measures are drawn through blue points for every pair of the sparsity regularization parameters $\left(\lambda^L, \lambda^G\right)$ considered in the S-ORRT construction. The values for CART and RF are depicted with a green diamond and an orange cross, respectively. It can be seen that S-ORRT outperforms CART in both prediction accuracy and global sparsity for several pairs of $\left(\lambda^L, \lambda^G\right)$. With respect to RF, S-ORRT is comparable in terms of prediction accuracy, while improving global sparsity in 50%.

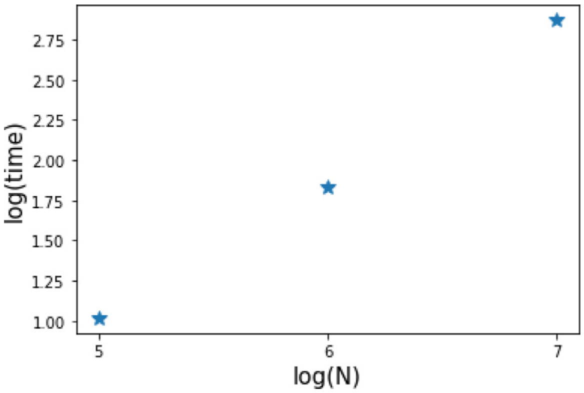

Fig. 5. Scalability of S-ORRT in logarithmic scale, where the computing time is measured in seconds as a function of N varying in $\left\{10^5, 10^6, 10^7\right\}$.

4.4. Scalability depending on the number of individuals: a simulation study

In this section we illustrate that S-ORRT scales up well with the size of the training sample N. To this aim, we measure the computing time taken by S-ORRT to reach a solution with 30% improvement on the mean squared error of CART.

We have designed a synthetic data set with $p = 25$ predictor variables and N taking values in $\left\{10^5, 10^6, 10^7\right\}$. The first two predictor variables, X_1 and X_2, define two balanced groups of individuals. They are generating following bivariate normal distributions, $\mathcal{N}(\boldsymbol{\eta}_k, \boldsymbol{\Sigma}_k)$, $k = 1, 2$.

$$\boldsymbol{\eta}_1 = (0.50, 0.75)^\top \text{ and } \boldsymbol{\Sigma}_1 = \begin{pmatrix} 0.005 & 0 \\ 0 & 0.00375 \end{pmatrix} \text{ for Group 1,}$$

and $\boldsymbol{\eta}_2 = (0.25, 0.50)^\top$ and $\boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_1$ for Group 2. The remaining 23 predictor variables were generated following a uniform distribution, $\mathcal{U}(0, 1)$. The response variable for Group 1 is equal to $Y = X_3 + 2X_4 + 5 + \varepsilon$ while for Group 2 is equal to $Y = -X_5 - 2X_6 - 5 + \varepsilon$, where $\varepsilon \sim \mathcal{N}(0, 0.5)$. Thus, $X_7, \ldots, X_{25}$ have no impact in the response variable. An S-ORRT tree of depth $D = 1$ with $\lambda^L = \lambda^G = 0$ is built. We feed Problem (3)–(5) with an initial solution, obtained from a heuristic procedure based on the RF variable importance measure. That is, in a first step we solve Problem (3)–(5) with a multistart approach in which the predictor variables with low RF variable importance, namely $X_j, \ j = 3, \ldots, 25$, do not play a role. This heuristic solution is given as the initial one to solve Problem (3)-(5) with the whole set of predictor variables. The procedure has been repeated 10 times and average results are presented. Figure 5 shows, as a function of N, the total computing time spent for the whole procedure. Both axes are on logarithmic scale. We can see that for this simulation study, the computing times have a linear trend with respect to the number of individuals.

5. Conclusions and future research

In recent years, several papers have focused on building decision trees in which the greedy suboptimal construction approach is replaced by solving an optimization problem, usually in integer variables. In this paper, we have adapted the continuous optimization-based approach to build classification trees previously proposed by the authors to consider regression trees. Local explanations on the continuous predictor space can be derived thanks to the smoothness of the predictions. Unlike CART and RF, we can directly model desirable properties such as sparsity, cost-sensitivity and fairness. The computational experience reported shows that our method outperforms CART, as well as OLS and LASSO, in terms of prediction accuracy. Finally, we show that our approach scales up well when the size of the training sample grows.

Several extensions to our approach are attractive. First, the linear prediction made at each leaf node can easily be extended to a non-linear one. This would be obtained by simply replacing the linear functions φ_{it} with other functions, such as those in a Generalized Additive Model. Second, it is known that standard Regression Analysis seeks an estimate of the conditional mean of the response variable, given the predictor vector, which is found by minimizing the mean squared error, as proposed in this paper. Nevertheless, it would be interesting to infer other characteristics of the distribution of the response variable, such as the conditional quantiles, with the final goal to obtain prediction intervals. An appropriate setting of our approach that considers quantile regression (Meinshausen, 2006) requires a nontrivial design. Third, a bagging scheme of our approach, where the collection of trees is solved simultaneously in order to have a global control on sparsity, is also an interesting open question. A parallelization framework would be suitable to make the training of a collection of trees tractable.

Acknowledgments

This research has been financed in part by research projects EC H2020 MSCA RISE NeEDS (Grant agreement ID: 822214), FQM-329 and P18-FR-2369 (Junta de Andalucía, Spain), and PID2019-110886RB-I00 (Ministerio de Ciencia e Innovación/Agencia Estatal de Investigación/10.13039/501100011033, Spain). This support is gratefully acknowledged.

Proof of Corollary 1. Observe that with $\boldsymbol{a}^* = \mathbf{0}$, by construction, $P_{it}\left(\mathbf{0}, \boldsymbol{\mu}_B^*\right)$ is independent of i. Hence, $\boldsymbol{P}\left(\mathbf{0}, \boldsymbol{\mu}_B^*\right)$ is a matrix with all its rows identical to a vector $u\left(\boldsymbol{\mu}_B^*\right)$, with $\sum_{t\in\tau_L} u_t\left(\boldsymbol{\mu}_B^*\right) = 1$.

Moreover, $\boldsymbol{\eta}\left(\mathbf{0}, \boldsymbol{\mu}_B^*\right) = -(y_i)_{i\in\mathcal{I}}$, and thus, for $\boldsymbol{a}^* = \mathbf{0}$ and $\boldsymbol{\mu}_B^*$, the vector $\boldsymbol{\mu}_L^* = \begin{pmatrix} -\bar{y}, & \cdots, & -\bar{y} \end{pmatrix}^\top$, satisfies the system of linear equations in Proposition 1. □

Proof of Proposition 2. First, let us consider the necessary optimality conditions for $\boldsymbol{a}^*$. For $\lambda^L \geq \lambda^l$ and $\lambda^G \geq \lambda^g$, we have, by construction, that

$$\begin{aligned} \lambda^L &\geq (1-\sigma)\left\|\nabla_{\boldsymbol{a}_{j\cdot}} \mathrm{MSE}(\mathbf{0}, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I})\right\|_\infty, \forall j = 1, \ldots, p, \\ \lambda^G &\geq \quad \sigma \quad \left\|\nabla_{\boldsymbol{a}_{j\cdot}} \mathrm{MSE}(\mathbf{0}, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I})\right\|_1, \forall j = 1, \ldots, p. \end{aligned}$$

Hence,

$$\begin{aligned} &-(1-\sigma)\nabla_{\boldsymbol{a}_{j\cdot}} \mathrm{MSE}(\boldsymbol{a}^*, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I}) \in \lambda^L \partial_{\boldsymbol{a}_{j\cdot}}\left(\left\|\boldsymbol{a}_{j\cdot}\right\|_1\right)\Big|_{\boldsymbol{a}_{j\cdot}=\mathbf{0}}, \\ &\quad \forall j = 1, \ldots, p, \\ &-\sigma\nabla_{\boldsymbol{a}_{j\cdot}} \mathrm{MSE}(\boldsymbol{a}^*, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I}) \in \lambda^G \partial_{\boldsymbol{a}_{j\cdot}}\left(\left\|\boldsymbol{a}_{j\cdot}\right\|_\infty\right)\Big|_{\boldsymbol{a}_{j\cdot}=\mathbf{0}}, \\ &\quad \forall j = 1, \ldots, p, \end{aligned}$$

and thus,

$$\begin{aligned} &-\nabla_{\boldsymbol{a}_{j\cdot}} \mathrm{MSE}(\boldsymbol{a}^*, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I}) \in \lambda^L \partial_{\boldsymbol{a}_{j\cdot}}\left(\left\|\boldsymbol{a}_{j\cdot}\right\|_1\right)\Big|_{\boldsymbol{a}_{j\cdot}=\mathbf{0}} \\ &+ \lambda^G \partial_{\boldsymbol{a}_{j\cdot}}\left(\left\|\boldsymbol{a}_{j\cdot}\right\|_\infty\right)\Big|_{\boldsymbol{a}_{j\cdot}=\mathbf{0}}, \forall j = 1, \ldots, p, \end{aligned}$$

having that,

$$\begin{aligned} &-\nabla_{\boldsymbol{a}} \mathrm{MSE}(\boldsymbol{a}^*, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I}) \in \lambda^L \partial_{\boldsymbol{a}}\left(\sum_{j=1}^p \left\|\boldsymbol{a}_{j\cdot}\right\|_1\right)\Bigg|_{\boldsymbol{a}=\boldsymbol{a}^*} \\ &+\lambda^G \partial_{\boldsymbol{a}}\left(\sum_{j=1}^p \left\|\boldsymbol{a}_{j\cdot}\right\|_\infty\right)\Bigg|_{\boldsymbol{a}=\boldsymbol{a}^*}, \end{aligned}$$

i.e.:

$$\mathbf{0} \in \partial_{\boldsymbol{a}}\left(\mathrm{MSE}(\boldsymbol{a}^*, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*); \mathcal{I}) + \lambda^L \sum_{j=1}^p \left\|\boldsymbol{a}_{j\cdot}\right\|_1 + \lambda^G \sum_{j=1}^p \left\|\boldsymbol{a}_{j\cdot}\right\|_\infty\right)\Bigg|_{\boldsymbol{a}=\boldsymbol{a}^*}.$$

For $\boldsymbol{a}^* = \mathbf{0}$, Corollary 1 shows that the chosen $\boldsymbol{\mu}_L^*$ minimizes MSE, and is thus optimal for Problem (1). In consequence, $\boldsymbol{\mu}_L^*$ satisfies the necessary optimality conditions.

Finally, let us analyze the optimality conditions for $\boldsymbol{\mu}_B = \boldsymbol{\mu}_B^*$. Observe that

$$\nabla_{\boldsymbol{\mu}_B} \mathrm{MSE}(\boldsymbol{a}, (\boldsymbol{\mu}_B, \boldsymbol{\mu}_L)) = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} 2(\varphi_i(\boldsymbol{a}, \boldsymbol{\mu}) - y_i)\nabla_{\boldsymbol{\mu}_B}\varphi_i(\boldsymbol{a}, \boldsymbol{\mu}).$$

Since $\boldsymbol{a}^* = \mathbf{0}$, $\nabla_{\boldsymbol{\mu}_B}\varphi_i(\boldsymbol{a}^*, \boldsymbol{\mu}^*)$ does not depend on $i \in \mathcal{I}$, say $\nabla_{\boldsymbol{\mu}_B}\varphi_i(\boldsymbol{a}^*, \boldsymbol{\mu}^*) = v$ for all $i \in \mathcal{I}$. Hence,

$$\nabla_{\boldsymbol{\mu}_B} \mathrm{MSE}(\boldsymbol{a}^*, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*)) = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} 2(\varphi_i(\boldsymbol{a}^*, \boldsymbol{\mu}^*) - y_i)v.$$

By Corollary 1, $\varphi_i(\boldsymbol{a}^*, \boldsymbol{\mu}^*) = \bar{y}$, and thus $\sum_{i\in\mathcal{I}} (\varphi_i(\boldsymbol{a}^*, \boldsymbol{\mu}^*) - y_i) = 0$, implying

$$\nabla_{\boldsymbol{\mu}_B} \mathrm{MSE}(\boldsymbol{a}^*, (\boldsymbol{\mu}_B^*, \boldsymbol{\mu}_L^*)) = 0,$$

and the desired result follows. □

References

Aghaei, S., Azizi, M., & Vayanos, P. (2019). Learning optimal and fair decision trees for non-discriminative decision-making. In *Proceedings of the AAAI conference on artificial intelligence: vol. 33* (pp. 1418–1426).

Athey, S. (2018). The impact of machine learning on economics. In *The economics of artificial intelligence: An agenda*. University of Chicago Press.

Baesens, B., Setiono, R., Mues, C., & Vanthienen, J. (2003). Using neural network rule extraction and decision tables for credit-risk evaluation. *Management Science, 49*(3), 312–329.

Bennett, K. P., & Blue, J. (1996). Optimal decision trees. *Rensselaer Polytechnic Institute Math Report 214*.

Bertsimas, D., Dunn, J., & Paschalidis, A. (2017). Regression and classification using optimal decision trees. In *Undergraduate research technology conference (URTC), 2017 IEEE MIT* (pp. 1–4).

Better, M., Glover, F., & Samorani, M. (2010). Classification by vertical and cutting multi-hyperplane decision tree induction. *Decision Support Systems, 48*(3), 430–436.

Biau, G., & Scornet, E. (2016). A random forest guided tour. *TEST, 25*(2), 197–227.

Blanquero, R., Carrizosa, E., Molero-Río, C., & Romero Morales, D. (2020). Sparsity in optimal randomized classification trees. *European Journal of Operational Research, 284*(1), 255–272.

Blanquero, R., Carrizosa, E., Molero-Río, C., & Romero Morales, D. (2021a). Optimal randomized classification trees. *Computers & Operations Research, 132*, 105281.

Blanquero, R., Carrizosa, E., Ramírez-Cobo, P., & Sillero-Denamiel, M. R. (2021b). A cost-sensitive constrained lasso. *Advances in Data Analysis and Classification, 15*, 121–158.

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

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Carrizosa, E., Martín-Barragán, B., & Romero Morales, D. (2011). Detecting relevant variables and interactions in supervised classification. *European Journal of Operational Research, 213*(1), 260–269.

Carrizosa, E., Molero-Río, C., & Romero Morales, D. (2021). Mathematical optimization in classification and regression trees. *TOP, 29*(1), 5–33.

Chikalov, I., Hussain, S., & Moshkov, M. (2018). Bi-criteria optimization of decision trees with applications to data analysis. *European Journal of Operational Research, 266*(2), 689–701.

Deng, H., & Runger, G. (2012). Feature selection via regularized trees. In *The 2012 international joint conference on neural networks (IJCNN)* (pp. 1–8). IEEE.

Deng, H., & Runger, G. (2013). Gene selection with guided regularized random forest. *Pattern Recognition, 46*(12), 3483–3489.

Dunn, J. (2018). *Optimal trees for prediction and prescription*. Massachusetts Institute of Technology Ph.D. thesis..

Fernández-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research, 15*(1), 3133–3181.

Firat, M., Crognier, G., Gabor, A., Hurkens, C., & Zhang, Y. (2019). Column generation based math-heuristic for classification trees. *Computers & Operations Research, 116*, 104866.

Freitas, A. (2014). Comprehensible classification models: A position paper. *ACM SIGKDD Explorations Newsletter, 15*(1), 1–10.

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

Genuer, R., Poggi, J.-M., Tuleau-Malot, C., & Villa-Vialaneix, N. (2017). Random forests for big data. *Big Data Research, 9*, 28–46.

Gevrey, M., Dimopoulos, I., & Lek, S. (2003). Review and comparison of methods to study the contribution of variables in artificial neural network models. *Ecological Modelling, 160*(3), 249–264.

Goodman, B., & Flaxman, S. (2017). European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine, 38*(3), 50–57.

Günlük, O., Kalagnanam, J., Li, M., Menickelly, M., & Scheinberg, K. (2021). Optimal decision trees for categorical data via integer programming. *Journal of Global Optimization, 81*, 233–260.

Harrison, D., Jr., & Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management, 5*(1), 81–102.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning* (2nd ed.). New York: Springer.

Hu, X., Rudin, C., & Seltzer, M. (2019). Optimal sparse decision trees. Advances in Neural Information Processing Systems.

Hyafil, L., & Rivest, R. L. (1976). Constructing optimal binary decision trees is NP–complete. *Information Processing Letters, 5*(1), 15–17.

Jones, E., Oliphant, T., Peterson, P. et al. (2001). SciPy: Open source scientific tools for Python.

Jung, J., Concannon, C., Shroff, R., Goel, S., & Goldstein, D. G. (2017). Simple rules for complex decisions. arXiv preprint arXiv:1702.04690.

Kraft, D. (1988). A software package for sequential quadratic programming. *Technical Report DFVLR-FB 88-28, DLR German Aerospace Center - Institute for Flight Mechanics, Köln, Germany*.

Liaw, A., & Wiener, M. (2002). Classification and regression by randomforest. *R News, 2*(3), 18–22.

Lichman, M. (2013). UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences. http://archive.ics.uci.edu/ml.

Lundberg, S., Erion, G., Chen, H., DeGrave, A., Prutkin, J., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S. I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence, 2*(1), 2522–5839.

Lundberg, S., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in neural information processing systems* (pp. 4765–4774).

Martens, D., Baesens, B., Van Gestel, T., & Vanthienen, J. (2007). Comprehensible credit scoring models using rule extraction from support vector machines. *European Journal of Operational Research, 183*(3), 1466–1476.

Martín-Barragán, B., Lillo, R., & Romo, J. (2014). Interpretable support vector machines for functional data. *European Journal of Operational Research, 232*(1), 146–155.

Meinshausen, N. (2006). Quantile regression forests. *Journal of Machine Learning Research, 7*, 983–999.

Molnar, C., Casalicchio, G., & Bischl, B. (2020). Interpretable machine learning–a brief history, state-of-the-art and challenges. arXiv preprint arXiv:2010.09337.

Narodytska, N., Ignatiev, A., Pereira, F., Marques-Silva, J., & RAS, I. (2018). Learning optimal decision trees with SAT. In *Ijcai* (pp. 1362–1368).

Python Core Team (2015). Python: A dynamic, open source programming language. Python software foundation. https://www.python.org.

Ribeiro, M., Singh, S., & 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* (pp. 1135–1144).

Ridgeway, G. (2013). The pitfalls of prediction. *National Institute of Justice Journal, 271*, 34–40.

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.

Ruggieri, S. (2019). Complete search for feature selection in decision trees. *Journal of Machine Learning Research, 20*(104), 1–34.

Therneau, T., Atkinson, B., & Ripley, B. (2015). rpart: Recursive partitioning and regression trees. https://CRAN.R-project.org/package=rpart.

Tibshirani, R., Wainwright, M., & Hastie, T. (2015). *Statistical learning with sparsity. The lasso and generalizations*. Chapman and Hall/CRC.

Ustun, B., & Rudin, C. (2016). Supersparse linear integer models for optimized medical scoring systems. *Machine Learning, 102*(3), 349–391.

Verwer, S., & Zhang, Y. (2017). Learning decision trees with flexible constraints and objectives using integer optimization. In *International conference on AI and OR techniques in constraint programming for combinatorial optimization problems* (pp. 94–103). Springer.

Verwer, S., & Zhang, Y. (2019). Learning optimal classification trees using a binary linear program formulation. In *The thirty-third AAAI conference on artificial intelligence (AAAI-19): vol. 33* (pp. 1625–1632). AAAI Press.

Yang, L., Liu, S., Tsoka, S., & Papageorgiou, L. G. (2017). A regression tree approach using mathematical programming. *Expert Systems with Applications, 78*, 347–357.