Information Sciences 690 (2025) 121564

Contents lists available at ScienceDirect

Information Sciences

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

Tree smoothing: Post-hoc regularization of tree ensembles for interpretable machine learning

Bastian Pfeifer [a], Arne Gevaert [b,d], Markus Loecher [c,d], Andreas Holzinger [a,d,e,*]

[a] Institute of Medical Informatics, Statistics and Documentation, Medical University Graz, Austria
[b] Department of Applied Mathematics, Computer Science and Statistics, University of Ghent, Belgium
[c] Berlin School of Economy and Law, Berlin, Germany
[d] Human-Centered AI Lab, Institute of Forest Engineering, Department of Forest- and Soil Sciences, University of Natural Resources and Life Sciences Vienna, Austria
[e] xAI-Lab, Alberta Machine Intelligence Institute, University of Alberta, Edmonton, Canada

ARTICLE INFO

Keywords:
Explainable artificial intelligence
XAI
Random forests
Regularization
Feature importance

ABSTRACT

Random Forests (RFs) are powerful ensemble learning algorithms that are widely used in various machine learning tasks. However, they tend to overfit noisy or irrelevant features, which can result in decreased generalization performance. Post-hoc regularization techniques aim to solve this problem by modifying the structure of the learned ensemble after training. We propose a novel *post-hoc regularization via tree smoothing* for classification tasks to leverage the reliable class distributions closer to the root node whilst reducing the impact of more specific and potentially noisy splits deeper in the tree. Our novel approach allows for a form of pruning that does not alter the general structure of the trees, adjusting the influence of nodes based on their proximity to the root node. We evaluated the performance of our method on various machine learning benchmark data sets and on cancer data from The Cancer Genome Atlas (TCGA). Our approach demonstrates competitive performance compared to the state-of-the-art and, in the majority of cases, and outperforms it in most cases in terms of prediction accuracy, generalization, and interpretability.

1. Introduction and motivation

In the rapidly evolving fields of bioinformatics and biomedical informatics, the ability to efficiently and accurately analyze complex biological data is becoming paramount. Ensemble learning algorithms [5], particularly Random Forests (RFs) [6,28], have proven to be indispensable tools in this endeavor. Their robustness, interpretability, and exceptional performance in handling high-dimensional and nonlinear tabular data make them highly suitable for the application on complex biological data sets [4]. In this paper we introduce a novel method for post-hoc regularization via tree smoothing, called Beta-Binomial Tree Smoothing (BBTS), and we underscore the significance of RFs in facilitating research, enabling scientists to decode complex disease mechanisms and discover potential biomarkers, ultimately advancing our understanding of biology, health and disease.

RFs combine multiple decision trees to make predictions [17,2,16]. Decision trees are often used in the field of pattern recognition and classification due to their good interpretability and efficiency [15].

* Corresponding author at: Human-Centered AI Lab, Institute of Forest Engineering, Department of Forest- and Soil Sciences, University of Natural Resources and Life Sciences Vienna, Austria.
E-mail address: andreas.holzinger@human-centered.ai (A. Holzinger).

https://doi.org/10.1016/j.ins.2024.121564
Received 14 May 2024; Received in revised form 28 July 2024; Accepted 14 October 2024

Available online 16 October 2024

RFs play a crucial role in bioinformatics and biomedical informatics due to their exceptional ability to handle high-dimensional data and nonlinear relationships often encountered in biological data sets. Their robust ensemble learning framework enables effective feature selection, making them well-suited for genomics, proteomics, and other 'omics' data analyses. Additionally, the interpretability of RFs allows researchers to gain insights into complex biological processes, facilitating advancements in understanding disease mechanisms and identifying potential biomarkers [12].

Moreover, RFs exhibit scale invariance, making them particularly well-suited for heterogeneous data sets encountered in bioinformatics [14]. This attribute allows them to effectively handle features with varying scales, facilitating robust analysis across diverse biological data types and ensuring reliable model performance in the presence of mixed data modalities [29,26].

However, RFs can still suffer from overfitting, especially when the trees in the forest become highly complex and tailored to the training data.

In general, regularization is a fundamental technique that is used to prevent overfitting and enhance generalization performance by restricting the complexity of a model [3]. Post-regularization techniques aim to address this shortcoming by modifying or refining the RF model *after* the initial training phase. Post-regularization techniques for RFs refer to methods used to reduce overfitting and improve the generalization performance [10]. These techniques typically focus on adjusting the complexity of individual trees or applying ensemble-level modifications. Some commonly used post-regularization techniques for random forests include:

Pruning: Pruning involves removing unnecessary branches or nodes from individual trees to simplify their structure. This helps prevent overfitting and promotes better generalization by reducing the complexity of the trees [21]. Particularly, ensemble pruning is a widely used method for eliminating superfluous classifiers from a large ensemble. This technique not only decreases the overall resource consumption but also occasionally enhances the performance of the original ensemble. In the same vein, leaf-refinement is a method that enhances the performance of a tree ensemble by joint re-learning the probability estimates in the leaf nodes of the trees. This enables the use of smaller ensembles while maintaining their predictive capabilities [7].

Feature selection: RFs can sometimes include irrelevant or redundant features, which can degrade performance. Feature selection techniques aim to identify and remove such features from the model, allowing it to focus on the most informative ones and potentially reducing overfitting [19,27].

Calibration: Calibration techniques aim to refine the predicted probabilities of random forests to better align with the true class probabilities. This can be particularly useful in tasks where reliable probability estimates are important, such as in certain risk assessment or medical diagnosis scenarios [24,20].

Smoothing: Smoothing is used in Probability Estimation Trees (PETs). PETs are a generalization of a single decision tree by taking the observed frequencies at a leaf node as the class probability estimates for any test examples that fall into that leaf [34]. Hierarchical smoothing assumes that the class probabilities in the leaf nodes depend on the probabilities higher in the tree. M-branch smoothing, for instance, considers each node in the tree as a subsample of the upper parent. It smoothes the leaf node to its direct parent using M-estimation, with the parent also being smoothed recursively until the root node is reached. For more information see [34]. Another Smoothing-based method is Hierarchical Shrinkage [1]. We will discuss this method in more detail in Section 2.

The aforementioned post-regularization techniques provide various approaches to combat overfitting and enhance the generalization ability of tree ensembles. By incorporating these techniques into the random forest workflow, practitioners can often achieve better performance and more reliable and interpretable predictions on unseen data. This is especially crucial in the biomedical field [11]. Based on preliminary work [25], in the following we present a novel, simple but effective regularization technique for post-hoc calibration of the Decision trees' leaf node probabilities. Our method leverages the reliable class distributions closer to the root node, whilst potentially reducing the impact of more specific and potentially noisy splits deeper in the tree. By emphasizing a more generalizable approach, our work not only provides advances compared to the state-of-the-art through its interpretability, but also lays the foundation for future research and applications. The remainder of the paper is structured as follows: In Section 2, we provide a brief introduction to the Hierarchical Shrinkage method [1], which is closely related to our method. Next, we introduce Beta-Binomial Tree Smoothing in Section 3. In Section 4, we describe the data sets and evaluation strategy we will employ to evaluate the proposed method. The results are presented and discussed in Section 5; based on a set of benchmark data sets in Section 5.1, and with a real-world bioinformatics application on gene expression breast cancer data from the Cancer Genome Atlas in Section 5.3. We conclude and discuss possible directions for future work in Section 6.

2. Hierarchical shrinkage

Agarwal et al. (2022) [1] recently proposed a post-hoc regularization technique known as *Hierarchical Shrinkage* (HS). Without modifying the tree structure, HS recursively moves predictions in deeper nodes closer to their ancestors. Although this is effectively a form of smoothing as described in the previous section, the authors name this *Shrinkage*, as the method shrinks the *difference* between predictions of nodes and their parents. This additional regularization improves generalization performance and allows for smaller ensembles without sacrificing accuracy. HS also enhances post-hoc interpretations by reducing noise in feature importance measures, leading to more reliable and robust interpretations. The method replaces the average prediction of a leaf node with a weighted average of the mean responses of the leaf and its ancestors, controlled by a regularization parameter λ as defined in equation (2).

The following is a brief summary of the ideas proposed in [1]. Assume that we are given a training set $\mathcal{D}_n = (X; y)$. Our goal is to learn a tree model $\hat{f}$ that accurately represents the regression function based on this training data. Given a query point $\mathbf{x}$, let $t_L \subset t_{L-1} \subset \cdots \subset t_0$ denote its leaf-to-root path, with t_L and t_0 representing its leaf node and the root node respectively. For any node t, let $N(t)$ denote the number of samples it contains, and $\hat{\mathbb{E}}_t\{y\}$ the average response. The tree model prediction can be written as the telescoping sum

$$\hat{f}(\mathbf{x}) = \hat{\mathbb{E}}_{t_0}\{y\} + \sum_{l=1}^{L} \left(\hat{\mathbb{E}}_{t_l}\{y\} - \hat{\mathbb{E}}_{t_{l-1}}\{y\} \right) \tag{1}$$

HS transforms $\hat{f}$ into a shrunk model $\hat{f}_\lambda$ via the formula:

$$\hat{f}_\lambda(\mathbf{x}) := \hat{\mathbb{E}}_{t_0}\{y\} + \sum_{l=1}^{L} \frac{\hat{\mathbb{E}}_{t_l}\{y\} - \hat{\mathbb{E}}_{t_{l-1}}\{y\}}{1 + \lambda / N(t_{l-1})}, \tag{2}$$

where λ is a hyperparameter chosen by the user, for example by cross validation. HS maintains the tree structure, and only modifies the prediction over each leaf node.

3. Beta-binomial tree smoothing

3.1. Intuition of the proposed approach

The approach described here is inspired by the concept of Hierarchical Shrinkage, drawing parallels to pruning but without explicitly pruning the tree. Unlike traditional pruning, which reduces the tree complexity by decreasing depth, this method retains the overall structure of the trees and adjusts the probabilities at the leaf nodes. By giving more weight to the nodes near the root, the method aims to *smooth* the tree. Specifically, this approach involves updating a conjugate Beta prior $\mathbf{B}_{prior}(\alpha, \beta)$ from the root to the leaf nodes by subsequently adding the number of classified samples to the model parameters α (class 0) and β (class 1). The parameter of the Beta prior $\mathbf{B}_{prior}(\alpha, \beta)$ can be inferred based on training data and/or might consist of some human expert prior belief. A human-in-the-loop here can bring in knowledge about the prior belief, i.e. the parameters of the Beta prior $\mathbf{B}_{prior}(\alpha, \beta)$ can be influenced not only by training data but also by incorporating expert human knowledge. This knowledge injection allows for the inclusion of domain-specific insights as a domain expert often knows the prior beliefs, thereby enhancing the model's performance, reliability and robustness, consequently better reflect real-world conditions and expectations, thus enhance trust. The leaf node probabilities are determined using the probabilities of observing a specific class given the inferred posterior Beta distribution $\mathbf{B}_{posterior}(\alpha, \beta)$. A graphical illustration of our approach is shown in Fig. 1.

3.2. Mathematical formulation

We define a Beta prior distribution $\mathbf{B}_{prior}(\alpha, \beta)$ at the root node, and subsequently update the parameter of this prior along a path to the leaf node. The updating scheme of the conjugate prior is defined as follows:

$$\alpha(t_L) = \alpha_{prior} + \sum_{l=0}^{L} N_1(t_l) \tag{3}$$

$$\beta(t_L) = \beta_{prior} + \sum_{l=0}^{L} N_0(t_l), \tag{4}$$

where $N_0(t_l)$ refers to the number of samples labeled as class 0, and $N_1(t_l)$ is the number of samples labeled as class 1 at node t_l.

The leaf node probabilities are calculated as

$$\hat{f}_{\alpha,\beta}(\mathbf{x}) = \frac{\alpha(t_L)}{\alpha(t_L) + \beta(t_L)}. \tag{5}$$

An alternative formulation can be found in the Appendix in eq. (15). From the Beta posterior $\mathbf{B}_{posterior}(\alpha, \beta)$ and the corresponding density distribution the actual confidence about the inferred leaf probabilities can be obtained. As a consequence, the overall interpretability of the classification is improved since the certainty about the inferred probability is provided by the distribution, e.g. through the density around the point estimate $\hat{f}_{\alpha,\beta}(\mathbf{x})$.

3.3. Feature importance

The mean decrease in impurity (MDI) specifies the importance of a given feature. According to MDI, the impurity decrease at node t_l for a split on variable X is:

$$\mathcal{I}_\nabla(X, t_l) = \mathcal{I}(t_l) - \left(\frac{N_{\text{left},t_l}}{N_{t_l}} \cdot \mathcal{I}(\text{left}, t_l) + \frac{N_{\text{right},t_l}}{N_{t_l}} \cdot \mathcal{I}(\text{right}, t_l) \right), \tag{6}$$

where *left* and *right* refer to the child nodes resulting from the split on variable X at node t_l. The default impurity for a node in trees is simply the variance of a binomial or multinomial process: $\mathcal{I}(t_l) = 1 - \sum_{j=1}^{K} \hat{p}_j^2$, where $\hat{p}_j$ is the empirical node probability of class j.

For our BBTS scheme, we modify the impurity in a node t_l as

$$\mathcal{I}(t_l) = 1 - \tilde{p}_0(t_l)^2 - \tilde{p}_1(t_l)^2, \tag{7}$$

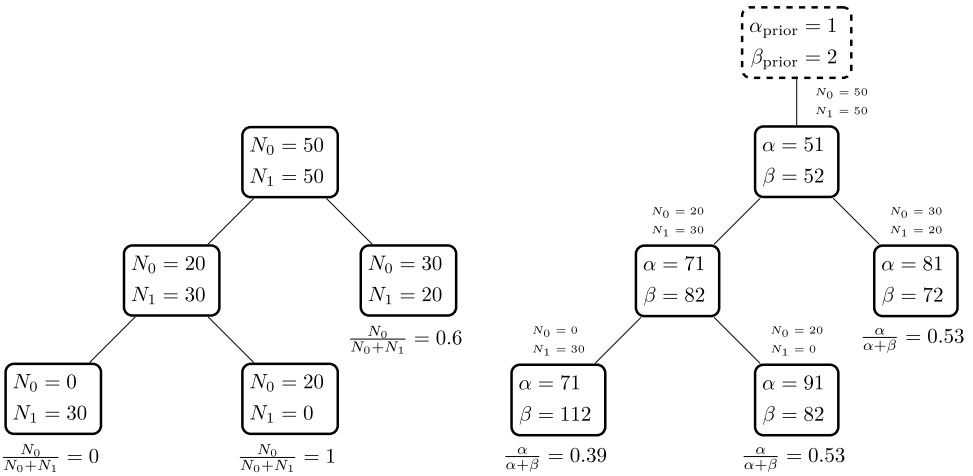

Fig. 1. Graphical representation of Beta-Binomial Tree Smoothing (BBTS). Left: classical decision tree. The predicted probability is the empirical average of the samples in the leaf. Right: BBTS. Each node represents a Beta distribution. Parameters α and β are updated using the label counts for each node. The predicted probability is the average of the Beta distribution in the leaf node.

where

$$\tilde{p}_0(t_l) = N_0(t_l)/(\alpha(t_l) + \beta(t_l)) \tag{8}$$

$$\tilde{p}_1(t_l) = N_1(t_l)/(\alpha(t_l) + \beta(t_l)). \tag{9}$$

We are using the actual node-specific sample counts $N_0(t_l)$ and $N_1(t_l)$ in the numerator, instead of $\alpha(t_l)$ and $\beta(t_l)$, because we do not want to measure the signal from an entire path to the node, but instead aim to measure the *local* effect the feature has on the *global* Beta-binomial distribution.

The MDI for variable X at node t_l in a specific tree t is defined as:

$$\text{MDI}^{(t)}(X) = \sum_{x \in t_l} \mathcal{I}_{\nabla}(X, t_l), \tag{10}$$

where $x \in t_l$ refers to a node in tree t a specific variable X was used for splitting. This is especially relevant when feature sampling is performed with replacement.

The MDI for variable X across all trees is:

$$\text{MDI}(X) = \frac{1}{T} \sum_{t=1}^{T} \text{MDI}^{(t)}(X) \tag{11}$$

The above equations define MDI with a modified impurity calculation, taking into account the α and β model parameter, so it aligns with the herein proposed tree smoothing post-regularization technique. We refer the reader to section 5.2 for insights on this new definition of MDI on simulated data.

3.4. Weighted Bayesian model averaging

Here, we outline a possible approach for aggregating the posterior distributions from specific leaf nodes decisive for test sample $\mathbf{y}$ to form a global ensemble-wide posterior distribution. This approach leverages Bayesian Model Averaging (BMA) to combine the individual tree-level information and obtain a more robust and uncertainty-aware prediction for a given test sample. The posterior distributions associated with the specific leaf nodes in each tree of the random forest are aggregated using BMA. For each leaf node in each tree, we denote the Beta distribution as $\mathbf{B}(\alpha_{i,\text{leaf}}, \beta_{i,\text{leaf}})$. The aggregated parameters for the ensemble of N decision trees are calculated as follows:

$$\alpha_{\text{BMA}} = \sum_{i=1}^{N} w_i \cdot \alpha_{i,\text{leaf}} \tag{12}$$

$$\beta_{\text{BMA}} = \sum_{i=1}^{N} w_i \cdot \beta_{i,\text{leaf}} \tag{13}$$

The ensemble-wide aggregated Beta distribution $\text{Beta}(\alpha_{\text{BMA}}, \beta_{\text{BMA}})$ is utilized to make the final prediction for a given test sample $\mathbf{y}$. The probability of belonging to class 1 is determined as:

$$P(\mathbf{y}|\mathbf{B}(\alpha_{\text{BMA}}, \beta_{\text{BMA}})) = \frac{\alpha_{\text{BMA}}}{\alpha_{\text{BMA}} + \beta_{\text{BMA}}} \tag{14}$$

Table 1
Benchmark data sets.

Datasets	Samples	Features	Class 0	Class 1
Breast cancer	286	9	196	81
Habermann	306	3	81	225
Heart	270	15	150	120
Diabetes	768	8	500	268
German credit	1000	20	300	700
Juvenile	3640	286	3153	487

The weights w_i in the aggregation process are assigned based on the chosen criteria, such as accuracy, performance, or biological relevance, reflecting the reliability or importance of each leaf node of a tree within the ensemble.

We have investigated the general applicability of the suggested BMA aggregation scheme while using the *true positive rate* as the weights w. However, in this paper, we put the focus on a comprehensive evaluation of the $\hat{f}_{\alpha,\beta}(\mathbf{x})$ estimates (equation (5)), which is the basis for the BMA approach.

3.5. Outlined similarities and differences to hierarchical shrinkage

[1] show a direct connection of their proposed Hierarchical Shrinkage method to *ridge regression* (in a special feature space) which often performs favorably in the bias-variance trade off especially at low sample sizes. From a Bayesian standpoint, ridge regression is equivalent to assuming a Gaussian with mean zero and variance $\tau^2 = \sigma^2/\lambda$[1] as the prior distribution for the regression coefficients. For each child node in a tree, its estimated mean response is shrunk to its parent node mean, hence the difference is shrunk toward zero. While the same one-parameter regularization can be applied to classification problems, a more natural prior (conjugate to the binomial distribution) is the Beta distribution which requires two hyperparameters α_{prior} and β_{prior} to be tuned.

Both HS and our Beta-Binomial Tree Smoothing (BBTS) are post-hoc procedures which do not alter the actual structure of the trees. They also both compute the sample size dependent shrinkage in a *hierarchical* fashion from parent node to child node unlike the superficially similar Leaf Based Shrinkage (LBS) [8] which shrinks each leaf directly towards the overall sample mean of the responses. A key difference between HS and our approach is that we compute a posterior distribution along the internal nodes down to the leaf nodes. As a consequence, the leaf nodes capture more information about the entire tree compared to HS.

4. Evaluation

4.1. Data sets

We assessed the accuracy of our post-hoc regularization technique on six machine learning benchmark data sets (see Table 1). The data sets were retrieved using the Python packages imodels [31] and PMLB [30]. Moreover, we put special focus on the evaluation of our method on gene expression breast cancer data from The Cancer Genome Atlas (TCGA) (https://www.cancer.gov/). The corresponding evaluation set-up and the obtained results are reported in Section 6.

Moreover, in close analogy to [32], we used the following data generating process, which we revisit in section 5.2. A binary response variable Y is predicted from a set of 5 predictor variables that vary in their scale of measurement and number of categories. The first predictor variable X_1 is continuous $X_1 \sim \mathcal{N}(0, 1)$, while the other predictor variables $X_2, \ldots, X_5$ are multinomial with $2, 4, 10, 20$ categories, respectively. The sample size for all simulation studies was set to $n = 200$. The distribution of the response is a Bernoulli process with probabilities that depend on X_2 via a "*relevance*" parameter $r \in [0; 0.5]$ which is related to the signal-to-noise-ratio (SNR), namely $P(y = 1|X_2 = 1) = 0.5 - r, P(y = 1|X_2 = 2) = 0.5 + r$. We would naturally hope that a reasonable variable importance measure would assign non zero scores only to X_2.

4.2. Evaluation strategy

In the initial experiment, we used grid search-based on 5-fold cross validation to infer the optimal values for α_{prior} and β_{prior}. Following we performed 5-fold crossvalidation with the inferred hyperparameters and calculated the mean accuracy. The aforementioned procedure was repeated 20 times and we report on results based on balanced accuracy and ROC-AUC.

In a subsequent experiment, we partitioned the data into training (80%) and testing (20%) sets. On the train dataset 5-fold crossvalidation was performed to tune the hyperparameters. The tuned model was then tested on the independent test data set. The described procedure was repeated 20 times, including the train-test split, and we report the classification performance based on balanced accuracy and ROC-AUC.

In case of the herein *Tree Smoothing* technique, the best performing Beta-specific hyperparameters α_{prior} and β_{prior} were grid-searched within $[2000, 1000, 800, 500, 100, 50, 30, 10, 1]$. For the *Hierarchical Shrinkage* method we used $\lambda = [0.001, 0.01, 0.1, 1, 10, 25, 50, 100, 200]$ as proposed by [1].

[1] where σ^2 is the residual variance.

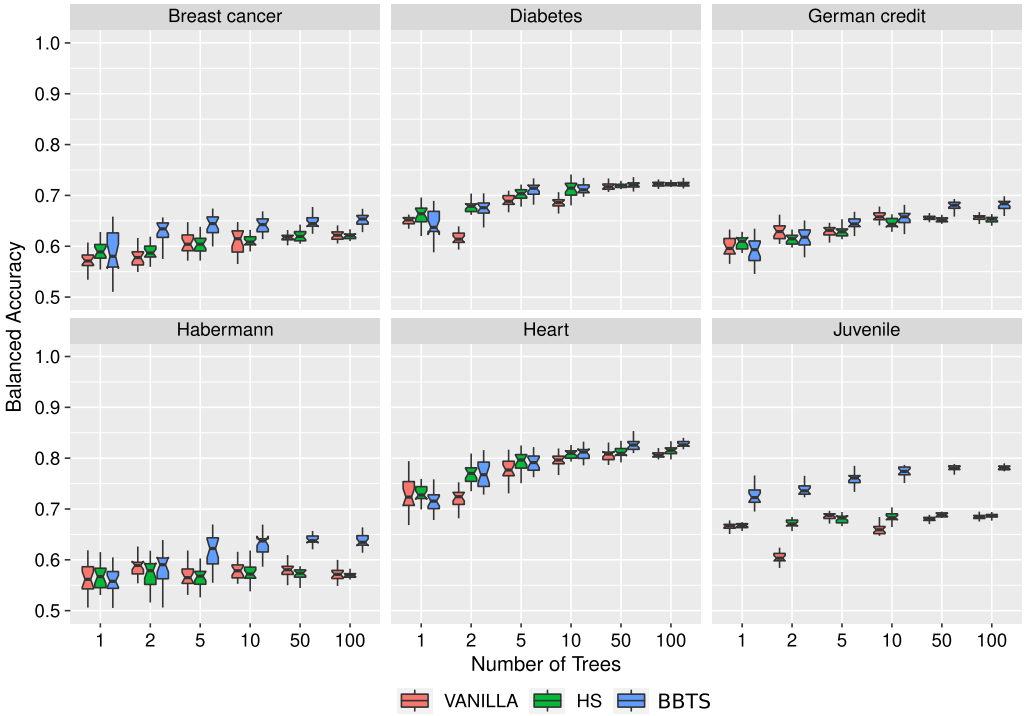

Fig. 2. *Balanced accuracy on six machine learning benchmark data sets.* The 5-fold cross-validation performance of Random Forest (vanilla RF), Hierarchical Shrinkage (HS), and our Beta-Binomial Tree Smoothing (BBTS) across varying number of trees within the ensemble is evaluated. The results are based on 20 train-test split iterations for each set-up.

In a subsequent experiment, we compared BBTS with a recently developed method called ForestPrune [21]. ForestPrune is based on the pruning concept and thus considerable different to the BBTS and HS approach, where the regularization is achieved by adopting the node probabilities instead of altering the structure of the tree ensemble.

5. Results and discussion

5.1. Benchmark data sets

The results based on the herein analyzed six benchmark data sets suggest that our approach is competitive with the Hierarchical Shrinkage method (HS) when ROC-AUC is used as a performance metric (see Fig. 3 and Fig. 5).

Clearly, post-regularization and hierarchical shrinkage improves the accuracy of the vanilla RF. In terms of balanced accuracy, however, our Tree Smoothing regularization procedure is superior in the majority of cases, when the median is utilized for comparison (see Fig. 2 and Fig. 4). The results based on the Juvenile data set highlight that our empirical Bayesian approach, enabling the prior to be inferred, could have a balancing effect on unbalanced class distributions (see Fig. 2, Fig. 4, and Table 1). In addition, we could learn that a flat prior $\mathbf{B}_{prior}(1,1)$ did not perform well in almost all cases, which suggests that the prior is essential for the calibration of the ensemble classifier.

Furthermore, we found that Tree Smoothing and Hierarchical Shrinkage can have a substantial influence when the number of trees within the ensemble are low. In case of a single decision tree, for instance, a positive effect can be observed for almost all benchmark data sets. This phenomenon sheds light on the nuanced relationship between regularization techniques and ensemble size, providing valuable insights for practitioners aiming to optimize computational resources and/or building more parsimony and thus interpretable models.

Interestingly, this effect is not as substantial when balanced accuracy is used as a performance metric. The observed phenomenon, where the impact of Tree Smoothing and Hierarchical Shrinkage is less substantial when using balanced accuracy as a performance metric, suggests an important interplay between the regularization techniques and the specific metrics employed.

The competitive performance of vanilla Random Forests in certain cases implies that the considerations captured by balanced accuracy differ from those emphasized by ROC-AUC, enriching our understanding of the methodological intricacies in evaluating classification models. This finding expands upon the ROC-AUC-centric approach in the work by [1], shedding light on the complex relationship between regularization techniques and metric choices. Overall, there is much more variance in the outcome in the case of the train-test split evaluation (Fig. 5 and Fig. 4) compared to the 5-fold cross-validation set-up (Fig. 3 and Fig. 2).

Moreover, the BMA aggregation scheme (equation (14)) had a positive effect on the test performance (see Fig. 10). In almost all cases we could observe an increase in performance using the probabilities of the weighted global posterior distribution.

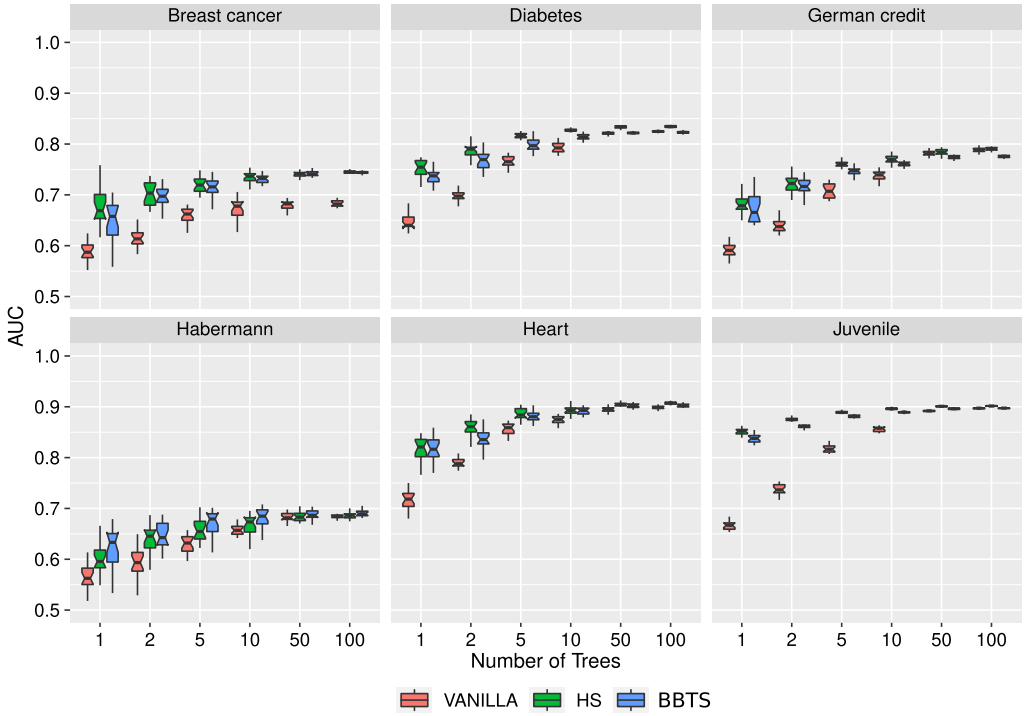

Fig. 3. *ROC-AUC on six machine learning benchmark data sets*. The 5-fold cross-validation performance of Random Forest (vanilla RF), Hierarchical Shrinkage (HS), and our Beta-Binomial Tree Smoothing (BBTS) across varying number of trees within the ensemble is evaluated. The results are based on 20 train-test split iterations for each set-up.

Finally, the experiments and comparison with ForestPrune (see Fig. 11) indicate that BBTS-based regularization is the better strategy when it comes to performance. However, the authors of ForestPrune explicitly state that they do not expect any increase in performance, but rather put their focus on decreasing the complexity of the tree ensemble while maintaining its performance.

5.2. Simulated data

Empirical studies have demonstrated that MDI often assigns higher importance values to high entropy features [32,22] or variables with high category frequencies. This phenomenon occurs because standard RFs may overfit the outcome variable using features that offer many potential splitting points. Therefore high MDI values do not always indicate strong predictive associations between features and the outcome.

This discrepancy is referred to as MDI "feature selection bias" and prominently displayed in Fig. 6. The vanilla MDI shows a strong preference for variables with many categories and the continuous variable and fails entirely to identify the relevant predictor variable. In fact, the mean value for the relevant variable X_2 is lowest and only slightly higher than in the null case.

Fig. 6 also indicates that our proposed technique accurately detects the relevant feature X_2. High entropy features tend to occur deeper in the trees and are down-weighted by our method. As a consequence, our approach detects the relevant feature already at a signal of $r = 0.15$, whereas in the case of the Hierarchical Shrinkage (HS) method there exists still some uncertainty about the actual ranking of the features.

5.3. Application on cancer data from the cancer genome atlas (TCGA)

In the field of biomedical informatics and bioinformatics we face huge amounts of data, often consisting of far more predictors p than samples n ($p >> n$). Random forests are fast, flexible and represent a robust approach to analyze high dimensional data [23]. In fact, RFs are still considered as state-of-the-art machine learning tools when these dimensions are organized in tabular form [16].

Furthermore, a key advantage over alternative machine learning algorithms are variable importance measures, which can be used to identify relevant features or perform variable selection [23]. In the biomedical domain, explainability is an important factor and critically important for clinicians in their daily practice [11,26].

We applied our methodology on gene expression data from human breast cancer patients. The data was retrieved from The Cancer Genome Atlas (TCGA) and was preprocessed as described in [9]. The resulting dataset comprised 981 patients and 8469 genes. The binary prediction task was to classify the samples into a group of patients with the luminal A subtype (499 samples) and patients with other breast cancer subtypes (482 samples).

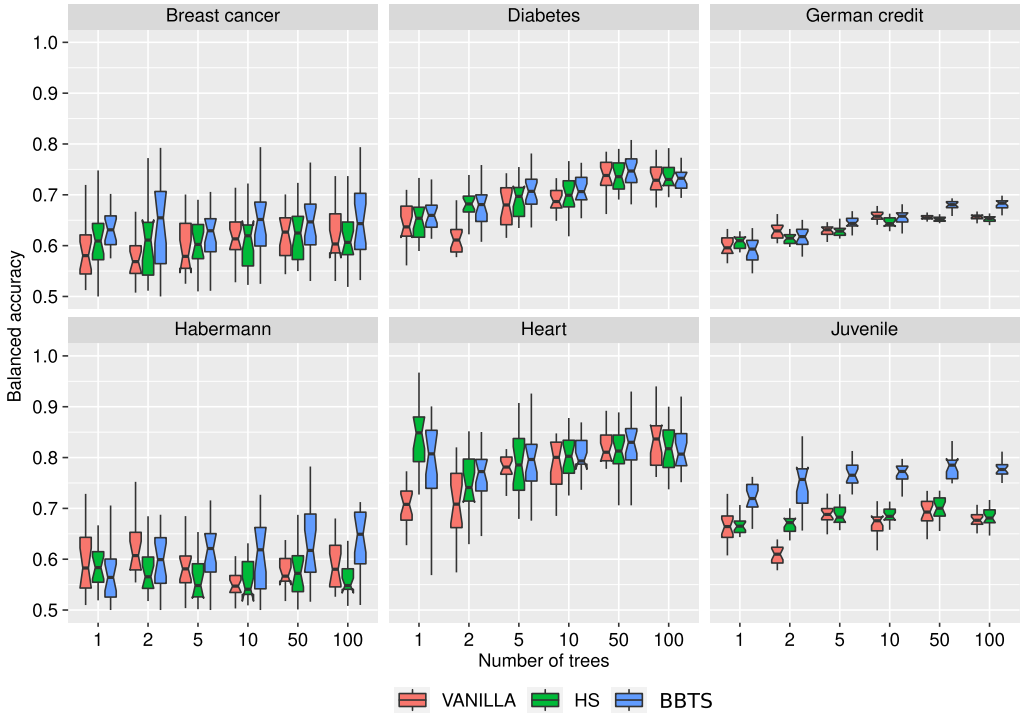

Fig. 4. *Balanced accuracy on six machine learning benchmark data sets.* The hold-out test set performance of Random Forest (vanilla RF), Hierarchical Shrinkage (HS), and our Beta-Binomial Tree Smoothing (BBTS) across varying number of trees within the ensemble is evaluated. The results are based on 20 train-test split iterations for each set-up.

Table 2
TCGA Breast Cancer data set.

Metric	n.trees	RF	HS	BBTS
BA	1	0.66	0.64	**0.70**
	2	0.64	0.69	**0.71**
	5	0.75	0.73	**0.80**
	10	0.75	0.75	**0.78**
	50	0.78	0.78	**0.80**
	100	0.78	0.78	**0.82**
AUC	1	0.65	**0.74**	0.73
	2	0.72	0.81	**0.83**
	5	0.81	0.85	**0.86**
	10	0.84	0.85	**0.87**
	50	0.88	0.88	**0.89**
	100	0.90	0.90	**0.91**

Display of the median hold-out test set performance values based on Balanced Accuracy (BA) and ROC-AUC out of 20 train-test split iterations. The best performing method is highlighted in bold.

The application to the TCGA breast cancer dataset demonstrate the superior performance of our method when balanced accuracy is used as a performance metric (see Fig. 7). Vanilla RF is competitive with Hierarchical Shrinkage with increasing numbers of trees within the ensemble. Our approach consistently produced higher accuracy values across varying sizes of the ensemble.

In case of ROC-AUC, both Hierarchical Shrinkage as well as our Beta-binomial regularization technique improve the performance of a vanilla RF. This is especially true when the number of trees within the ensemble is low. The obtained observation confirms the results in [1]. The authors pointed out the benefits of their approach on small number of trees. Still, post-hoc Tree Smoothing produced slightly better results, when the median ROC-AUC values are used for comparison (see Fig. 7 and Table 2).

Furthermore, we could observe that pruning by decreasing the maximal depth of the decision trees can have a positive effect on a vanilla random forest, however, almost has no influence on the post-regularization performance using HS or BBTS (see Fig. 12).

The competitive and often superior performance of our method on the TCGA breast cancer dataset, demonstrated through both ROC-AUC and balanced accuracy, highlights its effectiveness in biomedical real-world applications. These results contribute valuable insights into the nuanced behavior of the studied regularization techniques across different performance metrics and ensemble configurations.

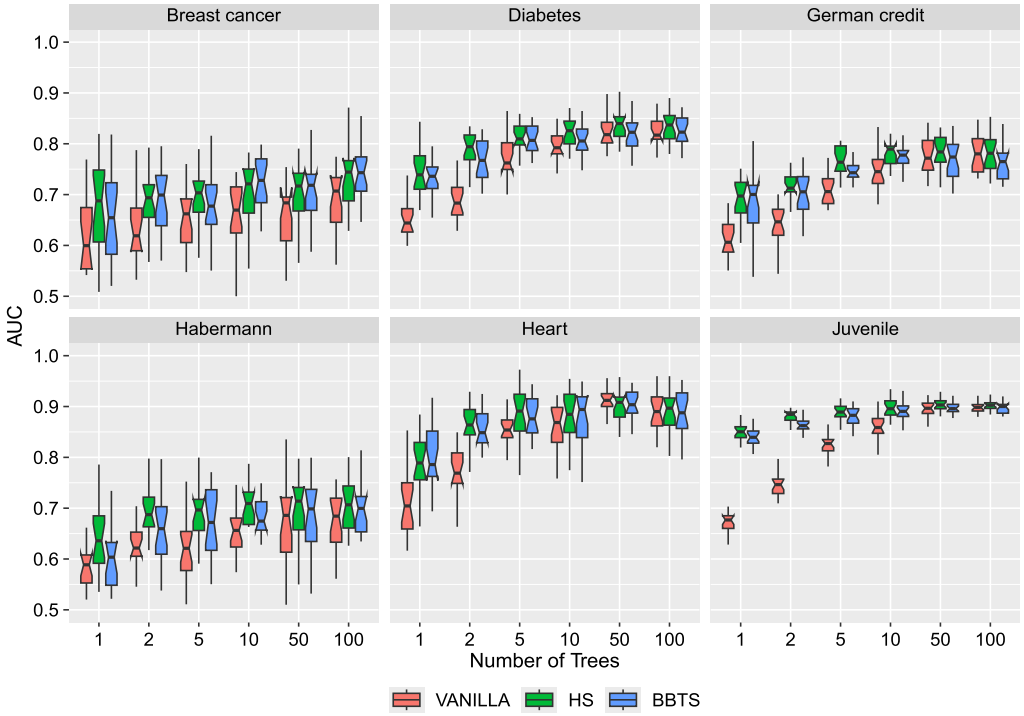

Fig. 5. *ROC-AUC on six machine learning benchmark data sets.* The hold-out test set performance of Random Forest (vanilla RF), Hierarchical Shrinkage (HS), and our Beta-Binomial Tree Smoothing (BBTS). The results are based on 20 train-test split iterations for each set-up.

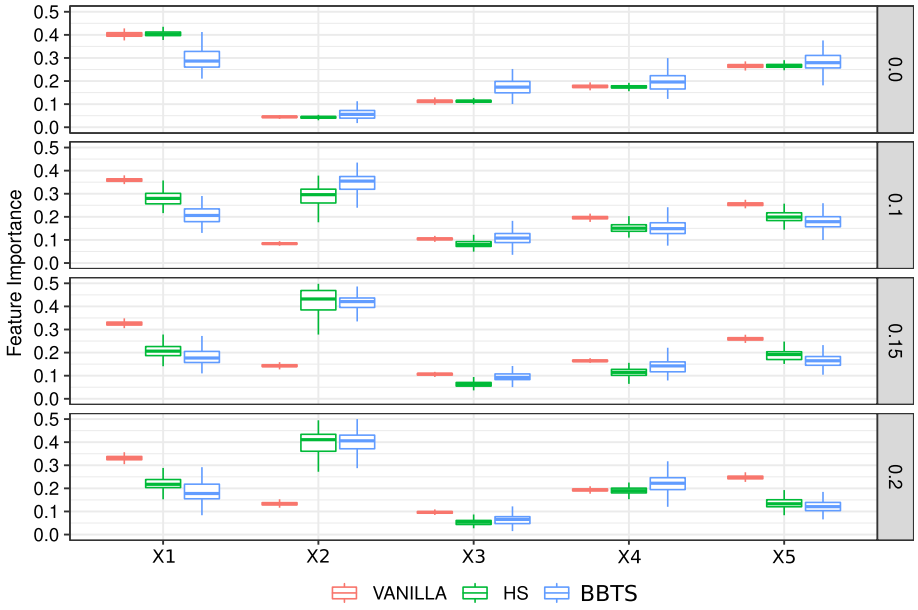

Fig. 6. Comparing MDI-based feature importance for various relevance r (SNR) values and shrinkage schemes. For low SNR the well-known bias towards features with high cardinality/entropy affects all approaches. For moderate to high SNR values, our Beta-binomial shrinkage separates the informative feature X_2 from the noisy features with the widest margins. The results are based on 100 iterations on the whole data set for each set-up.

Finally, to highlight the improved interpretability provided by our Tree Smoothing approach, we visualize the inferred distributions of the leaf nodes along the ensemble. In Fig. 8 we show the inferred α and β counts of the leaf nodes a single breast cancer patient was assigned to. The fact that the values within the leaf nodes are based on a well-defined model parameter allows us to derive additional conclusions. For instance, not only can the probability be obtained, but also the certainty of the model parameters. The density around the inferred parameter indicates to what extend we can trust the computed probabilities. A detailed inspection of the diagrams shown in Fig. 8 thus uncovers the trees and the corresponding leaf nodes for which a patient is classified with high certainty.

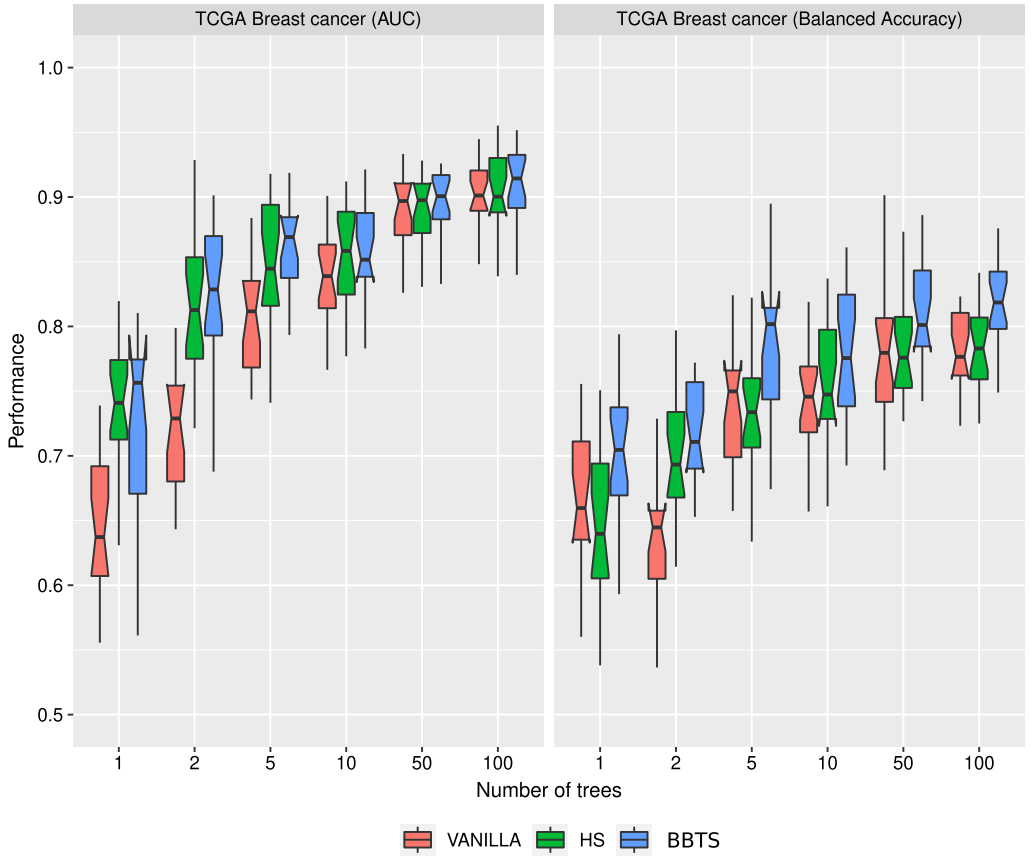

Fig. 7. *TCGA Breast Cancer dataset*. The hold-out test set classification performance of Random Forest (vanilla RF), Hierarchical Shrinkage (HS), and our Beta-Binomial Tree Smoothing (BBTS) while subsequently increasing the number of trees. The left panel shows the results based on ROC-AUC; the results based on balanced accuracy are displayed in the right panel. The results are based on 20 train-test split iterations for each set-up.

6. Conclusion and future outlook

We have introduced a novel post-regularization method for tree ensembles which we call Beta-Binomial Tree Smoothing (BBTS). BBTS significantly improves the accuracy of a vanilla random forest. Furthermore, we were able to show that our proposed method is particularly suitable for real-world biomedical data, such as gene expression profiles of cancer patients, where the features are continuous and often organized in a $p >> n$ setting.

By emphasizing a more general approach using empirical Bayes, our work not only provides advances but also lays the foundation for future research and applications. For example, the obtained posterior distribution in the leaf nodes, provides valuable information about the confidence and reliability of a prediction, which can help to further regularize and improve the ensemble classifier.

Future work may enhance similar techniques for regression tasks, for instance using conjugate Normal distributions, which can bring benefits in terms of computational simplicity because of straightforward analytical calculations of the posterior distribution. It can also support closed-form solutions as the posterior distribution parameters can be directly computed without resorting to numerical methods. Moreover, it can support the intuitive interpretation, because the resulting posterior distribution parameters provide clear insights into how the prior beliefs and observed data combine to form updated beliefs. Finally, the model-specific prior allows for incorporating domain-knowledge or expert-beliefs into the post-regularization procedure; a promising approach to further optimize the herein presented methodology, along with an investigation to understand how different types of expert knowledge influence the post-regularization process. Incorporating supplementary knowledge into data-driven machine learning models, a concept known as informed machine learning [33], is poised to gain prominence. This approach aims to enhance learning algorithms by integrating domain-specific information [18], thereby potentially improving model interpretability, robustness, and performance. As the field evolves, the synergy between explicit knowledge structures and machine learning could become a critical factor in developing advanced, human-centered AI systems particularly which enhance trust in the results [13].

CRediT authorship contribution statement

Bastian Pfeifer: Writing – review & editing, Writing – original draft, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Arne Gevaert:** Writing – original draft, Software, Methodology. **Markus Loecher:** Writing – review & editing,

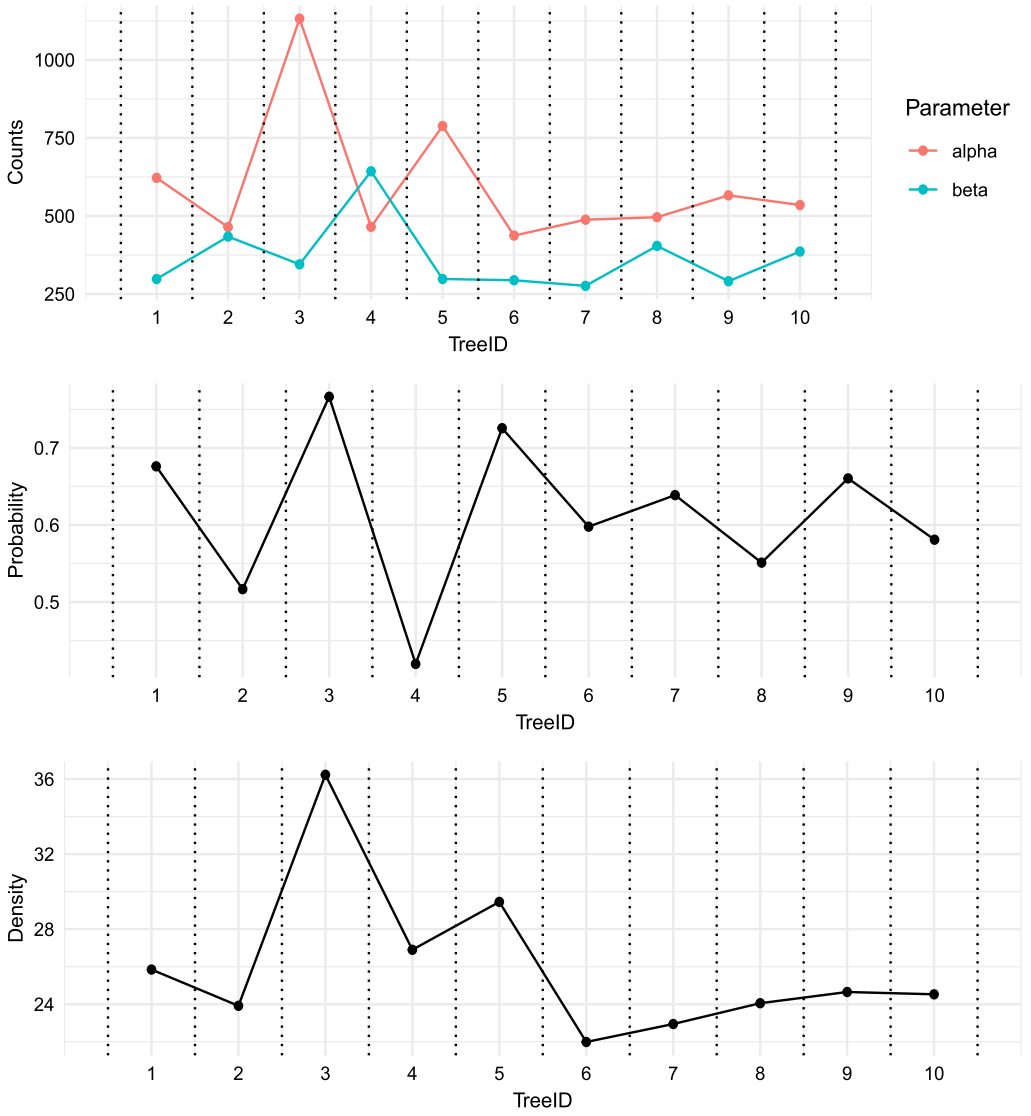

Fig. 8. *Classification of a single sample.* Shown are a set of values corresponding to the leaf node a sample was assigned to. In panel 1 the α and β values from the posterior distribution $\mathbf{B}_{posterior}(\alpha, \beta)$ are displayed. Panel 2 shows the corresponding $\hat{f}(\mathbf{x}; \alpha, \beta)$ values, and panel 3 draws the corresponding density values $\hat{f}(\mathbf{x}; \alpha, \beta)$ along the ten trees.

Writing – original draft, Supervision, Methodology, Investigation. **Andreas Holzinger:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Funding acquisition.

Software package `TreeSmoothing`

The proposed post-hoc regularization method as well as the Hierarchical Shrinkage technique is implemented within our Python package *TreeSmoothing*, which seemingly interfaces with sklearn functionalities and thus can be employed on any trained tree-based classifier. `TreeSmoothing` is available from PyPI (https://pypi.org/project/treesmoothing/) as well as from our GitHub repository (https://github.com/pievos101/TreeSmoothing).

Disclosure statement

During the preparation of this work the authors used Grammarly and DeepL in order to improve the English writing. After using this tools the authors reviewed and edited the content as needed and take full responsibility for the content of this publication.

Declaration of competing interest

The authors declare that they have no competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. This work does not raise any ethical issues.

Acknowledgements

Parts of this work have received funding from the Austrian Science Fund (FWF), Project: P-32554 (Explainable Artificial Intelligence); We thank Martin Urschler and Thomas Kuenzer for helpful discussions. The authors are grateful to the anonymous reviewers for their helpful comments.

Appendix A. Comparison to hierarchical shrinkage

The strong performance of the proposed Beta-Binomial Tree Smoothing (BBTS) across a wide variety of data sets is somewhat surprising in the light of the conventional wisdom that RFs excel when averaging many randomized, **deep** trees. In an attempt to shed light on why BBTS works so well, we show in this section that the smoothed leaf predictions from Eq. (5) can be written as a weighted sum of the actual node values and the prior mean:

$$\hat{f}_{\alpha,\beta}(\mathbf{x}) = \frac{\alpha(t_L)}{\alpha(t_L)+\beta(t_L)} = \frac{\alpha_{prior}+\sum_{l=0}^{L} N_1(t_l)}{\alpha_{prior}+\beta_{prior}+\sum_{l=0}^{L} N_1(t_l)+N_0(t_l)} \tag{15}$$

$$= w_p \cdot \frac{\alpha_{prior}}{\alpha_{prior}+\beta_{prior}} + \sum_{l=0}^{L} w_l \cdot \hat{\mathbb{E}}_{t_l}\{y\} \tag{16}$$

where $\hat{\mathbb{E}}_{t_l}\{y\} = N_1(t_l)/(N_1(t_l)+N_0(t_l))$ and the weights are defined as

$$w_p \equiv \frac{\alpha_{prior}+\beta_{prior}}{N_\Sigma}, \quad w_l \equiv \frac{N_1(t_l)+N_0(t_l)}{N_\Sigma} \tag{17}$$

and add up to 1, with the denominator denoted by the cumulative counts $N_\Sigma \equiv \alpha(t_L)+\beta(t_L) = \alpha_{prior}+\beta_{prior}+\sum_{l=0}^{L} N_1(t_l)+N_0(t_l)$. As the node counts necessarily diminish with each split, the weights w_l monotonically decrease from root to leaf.

An equivalent form of the HS scheme eq. (2) can also be written as a weighted sum of the actual node values:

$$\hat{f}_\lambda(\mathbf{x}) = \hat{\mathbb{E}}_0(1-S_0) + \sum_{l=1}^{L-1} \hat{\mathbb{E}}_l \cdot \left(S_{l-1}-S_l\right) + \hat{\mathbb{E}}_L \cdot S_{L-1}, \tag{18}$$

where this time the coefficients form a telescoping sum adding up to 1:

$$(1-S_0) + \sum_{l=1}^{L-1} \left(S_{l-1}-S_l\right) + S_{L-1} = 1 + (-S_0+S_0) + (-S_1+S_1) + \ldots + (-S_{L-1}+S_{L-1}) = 1 \tag{19}$$

with $S_l := \left(1+\lambda/N(t_l)\right)^{-1}$ and $\hat{\mathbb{E}}_l := \hat{\mathbb{E}}_{t_l}\{y\}$

Fig. 9 compares the BBTS weights with the weights from hierarchical shrinkage for the special case of a hypothetical balanced binary tree in which node counts halve at each split.

The differences are striking: for moderate values of λ, HS heavily weighs nodes in the middle of the tree and closer to the leaves, whereas BBTS appears to exponentially downscale node contributions with increasing depth.

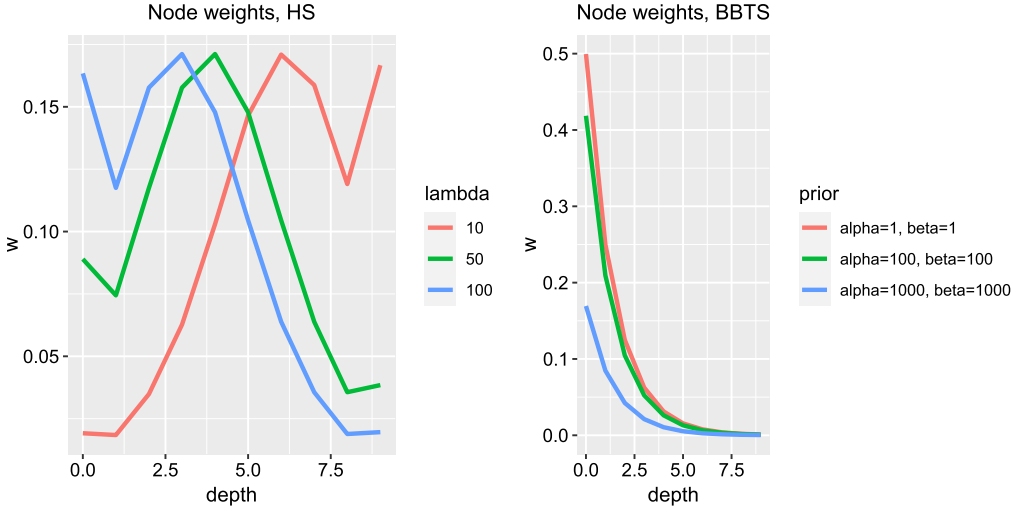

Fig. 9. Comparing node weights for Hierarchical Shrinkage (HS, left panel) and our BBTS scheme as defined in Eq. (17). For simplicity, we assume a perfect balanced binary tree of depth 8 in which node counts halve at each split and $N_1(t_l) = N_0(t_l), N_1(t_0) = 2^8$.

Appendix B. Influence of using Bayesian model averaging

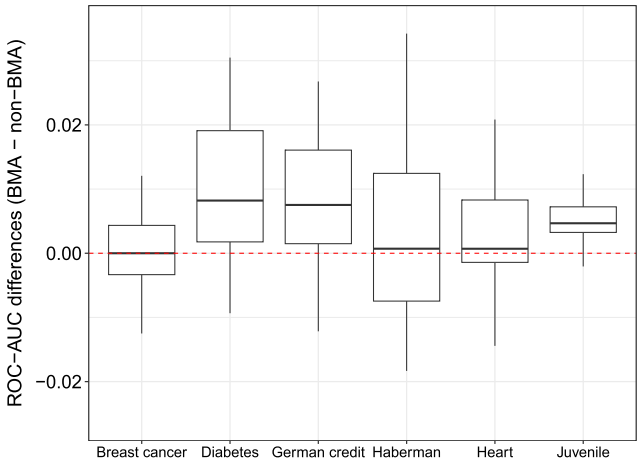

Fig. 10. *BMA versus non-BMA aggregation.* The differences in ROC-AUC values on six machine learning benchmark data sets are shown. The hold-out test set performance is based on 30 train-test split iterations for each set-up. The ensembles included 10 trees.

Appendix C. Comparison with ForestPrune

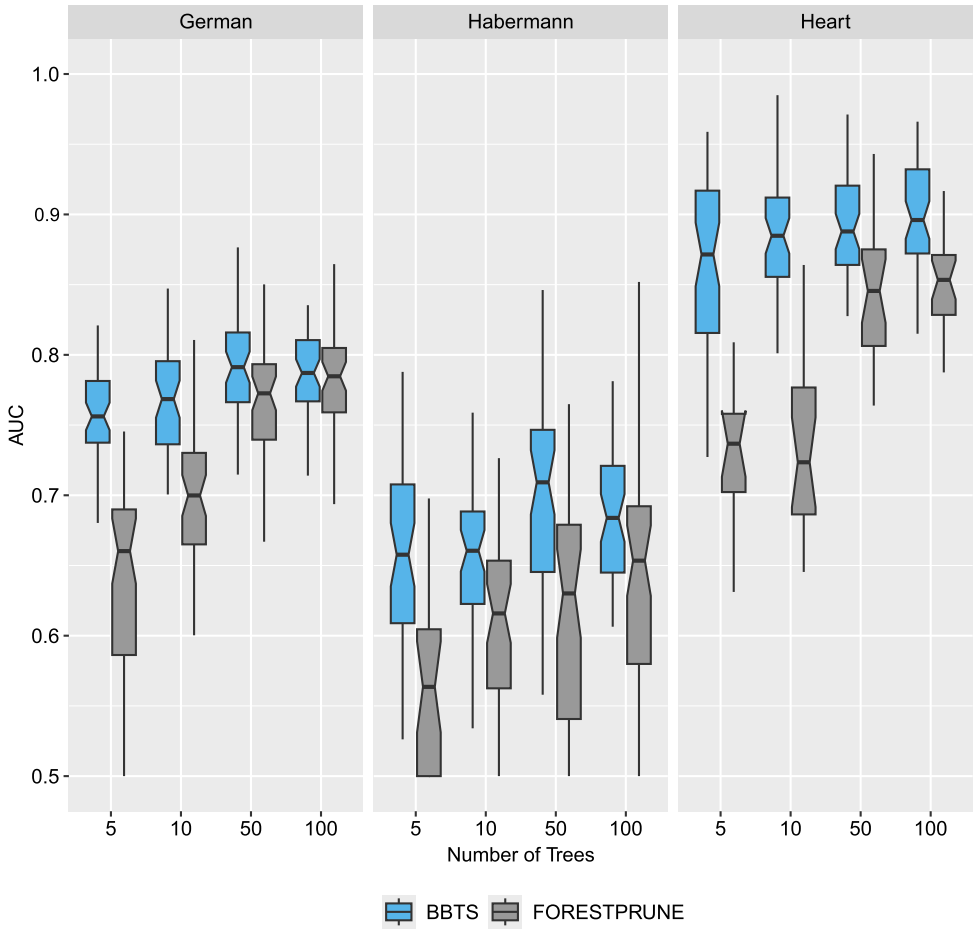

Fig. 11. *ROC-AUC on three machine learning benchmark data sets*. The hold-out test set performance of our Beta-Binomial Tree Smoothing (BBTS) and ForestPrune. The results are based on 50 train-test split iterations for each set-up.

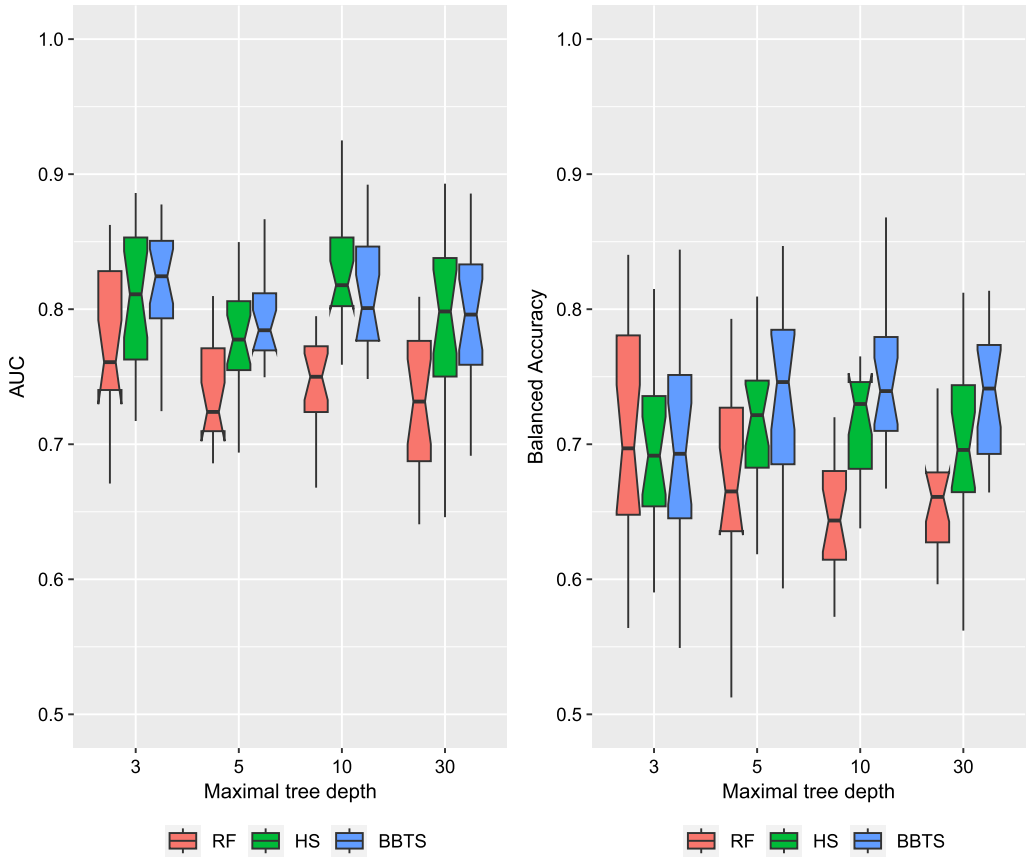

Fig. 12. *TCGA Breast Cancer dataset.* The hold-out test set classification performance of Random Forest (vanilla RF), Hierarchical Shrinkage (HS), and our Beta-Binomial Tree Smoothing (BBTS) while subsequently increasing the maximum tree depth. The left panel shows the results based on ROC-AUC; the results based on balanced accuracy are displayed in the right panel. The results are based on 20 train-test split iterations for each set-up. The number of trees is set to 2.

Appendix D. Glossary of notations

- AI = Artificial intelligence
- AUC = Area under the Curve
- BMA = Bayesian Model Averaging
- BBTS = Beta-Binominal Tree Smoothing
- $\mathbf{B}_{prior}(\alpha, \beta)$ = Beta prior distribution at the root node
- $\mathbf{B}_{posterior}(\alpha, \beta)$ = Beta posterior
- $\mathcal{D}_n$ = Training Set
- HS = Hierachical Shrinkage
- MDI = Mean Decrease in Impurity
- ML = Machine learning
- PET = Probability Estimation Trees
- PMLB = Penn Machine Learning Benchmarks
- RF = Random Forests
- ROC = Receiver Operating Characteristic
- TCGA = The Cancer Genome Atlas
- XAI = Explainable Artificial Intelligence

Data availability

Data will be made available on request.

References

[1] A. Agarwal, Y.S. Tan, O. Ronen, C. Singh, B. Yu, Hierarchical shrinkage: improving the accuracy and interpretability of tree-based models, in: International Conference on Machine Learning, Proceedings of Machine Learning Research (PMLR), 2022, pp. 111–135.

[2] G. Armano, E. Tamponi, Building forests of local trees, Pattern Recognit. 76 (4) (2018) 380–390, https://doi.org/10.1016/j.patcog.2017.11.017.
[3] R. Balestriero, L. Bottou, Y. LeCun, The effects of regularization and data augmentation are class dependent, Adv. Neural Inf. Process. Syst. 35 (2022) 37.
[4] L.G. Bernardini, C. Rosinger, G. Bodner, K.M. Keiblinger, E. Izquierdo-Verdiguier, H. Spiegel, C.O. Retzlaff, A. Holzinger, Learning vs. understanding: when does artificial intelligence outperform process-based modeling in soil organic carbon prediction?, New Biotechnol. 81 (7) (2024) 20–31, https://doi.org/10.1016/j.nbt.2024.03.001.
[5] V. Bolón-Canedo, A. Alonso-Betanzos, Ensembles for feature selection: a review and future trends, Inf. Fusion 52 (2019) 1–12, https://doi.org/10.1016/j.inffus.2018.11.008.
[6] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32, https://doi.org/10.1023/A:1010933404324.
[7] S. Buschjaeger, K. Morik, Joint leaf-refinement and ensemble pruning through l 1 regularization, Data Min. Knowl. Discov. 37 (3) (2023) 1230–1261, https://doi.org/10.1007/s10618-023-00921-z.
[8] T. Chen, C. Guestrin, Xgboost: a scalable tree boosting system, in: Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.
[9] H. Chereda, A. Bleckmann, K. Menck, J. Perera-Bel, P. Stegmaier, F. Auer, F. Kramer, A. Leha, T. Beißbarth, Explaining decisions of graph convolutional neural networks: patient-specific molecular subnetworks responsible for metastasis prediction in breast cancer, Gen. Med. 13 (2021) 1–16, https://doi.org/10.1186/s13073-021-00845-7.
[10] V. Chernozhukov, C. Hansen, M. Spindler, Valid post-selection and post-regularization inference: an elementary, general approach, Annu. Rev. Econ. 7 (1) (2015) 649–688, https://doi.org/10.1146/annurev-economics-012315-015826.
[11] C. Combi, B. Amico, R. Bellazzi, A. Holzinger, J.H. Moore, M. Zitnik, J.H. Holmes, A manifesto on explainability for artificial intelligence in medicine, Artif. Intell. Med. 133 (11) (2022) 102423, https://doi.org/10.1016/j.artmed.2022.102423.
[12] S. Dara, S. Dhamercherla, S.S. Jadav, C.M. Babu, M.J. Ahsan, Machine learning in drug discovery: a review, Artif. Intell. Rev. 55 (3) (2022) 1947–1999, https://doi.org/10.1007/s10462-021-10058-4.
[13] J. Del Ser, A. Barredo-Arrieta, N. Díaz-Rodríguez, F. Herrera, A. Saranti, A. Holzinger, On generating trustworthy counterfactual explanations, Inf. Sci. 655 (2024) 119898, https://doi.org/10.1016/j.ins.2023.119898.
[14] H. Deng, G. Runger, Gene selection with guided regularized random forest, Pattern Recognit. 46 (12) (2013) 3483–3489, https://doi.org/10.1016/j.patcog.2013.05.018.
[15] B. Gao, Q. Zhou, Y. Deng, HIE-EDT: hierarchical interval estimation-based evidential decision tree, Pattern Recognit. 146 (2024) 110,040, https://doi.org/10.1016/j.patcog.2023.110040.
[16] L. Grinsztajn, E. Oyallon, G. Varoquaux, Why do tree-based models still outperform deep learning on typical tabular data?, in: S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), Advances in Neural Information Processing Systems (NeurIPS 2022), Curran Associates, Inc., 2022, pp. 507–520.
[17] T.K. Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Anal. Mach. Intell. 20 (8) (1998) 832–844, https://doi.org/10.1109/34.709601.
[18] A. Holzinger, A. Saranti, A.C. Hauschild, J. Beinecke, D. Heider, R. Roettger, H. Mueller, J. Baumbach, B. Pfeifer, Human-in-the-loop integration with domain-knowledge graphs for explainable federated deep learning, in: Lecture Notes in Computer Science (LNCS), vol. 14065, Springer, 2023, pp. 45–64.
[19] M.B. Kursa, W.R. Rudnicki, Feature selection with the boruta package, J. Stat. Softw. 36 (11) (2010) 1–13, https://doi.org/10.18637/jss.v036.i11.
[20] T. Leathart, E. Frank, G. Holmes, B. Pfahringer, Probability calibration trees, in: Y.K. Noh, M.L. Zhang (Eds.), Asian Conference on Machine Learning, PMLR, 2017, pp. 145–160.
[21] B. Liu, R. Mazumder, ForestPrune: compact depth-pruned tree ensembles, in: F. Ruiz, J. Dy, J.W. van de Meent (Eds.), Proceedings of the 26th International Conference on Artificial Intelligence and Statistics, Proceedings of Machine Learning Research (PMLR), vol. 206, 2023, pp. 9417–9428.
[22] M. Loecher, Debiasing MDI feature importance and SHAP values in tree ensembles, in: International Cross-Domain Conference for Machine Learning and Knowledge Extraction, Springer, 2022, pp. 114–129.
[23] S. Nembrini, I.R. König, M.N. Wright, The revival of the Gini importance?, Bioinformatics 34 (21) (2018) 3711–3718, https://doi.org/10.1093/bioinformatics/bty373.
[24] A. Niculescu-Mizil, R. Caruana, Predicting good probabilities with supervised learning, in: Proceedings of the 22nd International Conference on Machine Learning (ICML 2005), 2005, pp. 625–632.
[25] B. Pfeifer, Bayesian post-hoc regularization of random forests, arXiv preprint, arXiv:2306.03702, 2023, https://doi.org/10.48550/arXiv.2306.03702.
[26] B. Pfeifer, H. Baniecki, A. Saranti, P. Biecek, A. Holzinger, Multi-omics disease module detection with an explainable greedy decision forest, Sci. Rep. 12 (1) (2022) 16857, https://doi.org/10.1038/s41598-022-21417-8.
[27] B. Pfeifer, A. Holzinger, M.G. Schimek, Robust random forest-based all-relevant feature ranks for trustworthy AI, Stud. Health Technol. Inform. 294 (2022) 137–138, https://doi.org/10.3233/SHTI220418.
[28] X. Qiu, L. Zhang, P.N. Suganthan, G.A. Amaratunga, Oblique random forest ensemble via least square estimation for time series forecasting, Inf. Sci. 420 (2017) 249–262, https://doi.org/10.1016/j.ins.2017.08.060.
[29] P.S. Reel, S. Reel, E. Pearson, E. Trucco, E. Jefferson, Using machine learning approaches for multi-omics data analysis: a review, Biotechnol. Adv. 49 (2021) 107,739, https://doi.org/10.1016/j.biotechadv.2021.107739.
[30] J.D. Romano, T.T. Le, W. La Cava, J.T. Gregg, D.J. Goldberg, P. Chakraborty, N.L. Ray, D. Himmelstein, W. Fu, J.H. Moore, Pmlb v1.0: an open source dataset collection for benchmarking machine learning methods, arXiv preprint, arXiv:2012.00058v2, 2021.
[31] C. Singh, K. Nasseri, Y.S. Tan, T. Tang, B. Yu, imodels: a python package for fitting interpretable models, J. Open Sour. Softw. 6 (61) (2021) 3192, https://doi.org/10.21105/joss.03192.
[32] C. Strobl, A.L. Boulesteix, A. Zeileis, T. Hothorn, Bias in random forest variable importance measures: illustrations, sources and a solution, BMC Bioinform. 8 (1) (2007) 1–21, https://doi.org/10.1186/1471-2105-8-25.
[33] L. Von Rueden, S. Mayer, K. Beckh, B. Georgiev, S. Giesselbach, R. Heese, B. Kirsch, J. Pfrommer, A. Pick, R. Ramamurthy, M. Walczak, J. Garcke, C. Bauckhage, J. Schuecker, Informed machine learning–a taxonomy and survey of integrating prior knowledge into learning systems, IEEE Trans. Knowl. Data Eng. 35 (1) (2021) 614–633, https://doi.org/10.1109/TKDE.2021.3079836.
[34] H. Zhang, F. Petitjean, W. Buntine, Hierarchical gradient smoothing for probability estimation trees, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2020, pp. 222–234.