From unbiased MDI Feature Importance to Explainable AI for Trees

Markus Loecher
Berlin School of Economics and Law,
10825 Berlin, Germany
markus.loecher@hwr-berlin.de

Key Words: variable importance; random forests; trees; Gini impurity; explainable AI.

ABSTRACT

We attempt to give a unifying view of the various recent attempts to (i) improve the interpretability of tree-based models and (ii) debias the the default variable-importance measure in random Forests, Gini importance. In particular, we demonstrate a common thread among the out-of-bag based bias correction methods and their connection to local explanation for trees. In addition, we point out a bias caused by the inclusion of inbag data in the newly developed SHAP values and suggest a remedy.

1 Variable importance in trees

Variable importance is not very well defined as a concept. Even for the case of a linear model with n observations, p variables and the standard $n >> p$ situation, there is no theoretically defined variable importance metric in the sense of a parametric quantity that a variable importance estimator should try to estimate (Grömping, 2009). Variable importance measures for random forests have been receiving increased attention in bioinformatics, for instance to select a subset of genetic markers relevant for the prediction of a certain disease. They also have been used as screening tools (Díaz-Uriarte and De Andres, 2006, Menze et al., 2009) in important applications highlighting the need for reliable and well-understood feature importance measures.

The default choice in most software implementations (Liaw and Wiener, 2002, Pedregosa et al., 2011) of random forests (Breiman, 2001) is the *mean decrease in impurity (MDI)*. The

MDI of a feature is computed as a (weighted) mean of the individual trees' improvement in the splitting criterion produced by each variable. A substantial shortcoming of this default measure is its evaluation on the in-bag samples which can lead to severe overfitting (Kim and Loh, 2001). It was also pointed out by Strobl et al. (2007a) that *the variable importance measures of Breiman's original Random Forest method ... are not reliable in situations where potential predictor variables vary in their scale of measurement or their number of categories.* There have been multiple attempts at correcting the well understood bias of the Gini impurity measure both as a split criterion as well as a contributor to importance scores, each one coming from a different perspective.

Strobl et al. (2007b) derive the exact distribution of the maximally selected Gini gain along with their resulting p-values by means of a combinatorial approach. Shih and Tsai (2004) suggest a solution to the bias for the case of regression trees as well as binary classification trees (Shih, 2004) which is also based on p-values. Several authors (Loh and Shih, 1997, Hothorn et al., 2006) argue that the criterion for split variable and split point selection should be separated.

A different approach is to add so-called pseudo variables to a dataset, which are permuted versions of the original variables and can be used to correct for bias (Sandri and Zuccolotto, 2008). Recently, a modified version of the Gini importance called Actual Impurity Reduction (AIR) was proposed Nembrini et al. (2018) that is faster than the original method proposed by Sandri and Zuccolotto with almost no overhead over the creation of the original RFs and available in the R package *ranger* (Wright and Ziegler, 2015, Wright et al., 2017).

2 Separating inbag and out-of-bag (oob) samples

An idea that is gaining quite a bit of momentum is to include OOB samples to compute a debiased version of the Gini importance (Li et al., 2019a, Zhou and Hooker, 2019, Loecher, 2020) yielding promising results. Here, the original Gini impurity (for node m) for a categorical variable Y which can take D values $c_1, c_2, \dots, c_D$ is defined as

$$G(m) = \sum_{d=1}^{D} \hat{p}_d(m) \cdot (1 - \hat{p}_d(m)), \text{ where} \hat{p}_d = \frac{1}{n_m} \sum_{i \in m} Y_i.$$

Loecher (2020) proposed a *penalized Gini impurity* which combines inbag and out-of-bag samples. The main idea is to increase the impurity $G(m)$ by a penalty that is proportional to the difference $\Delta = (\hat{p}_{OOB} - \hat{p}_{inbag})^2$:

$$PG_{oob}^{\alpha,\lambda} = \alpha \cdot G_{oob} + (1 - \alpha) \cdot G_{in} + \lambda \cdot (\hat{p}_{oob} - \hat{p}_{in})^2 \tag{1}$$

In addition, Loecher (2020) investigated replacing $G(m)$ by an unbiased estimator of the variance via the well known sample size correction.

$$\widehat{G}(m) = \frac{N}{N-1} \cdot G(m) \tag{2}$$

In this paper we focus on the following three special cases, $[\alpha = 1, \lambda = 2]$, $[\alpha = 0.5, \lambda = 1]$ as well as $[\alpha = 1, \lambda = 0]$:

$$PG_{oob}^{(1,2)} = \sum_{d=1}^{D} \hat{p}_{d,oob} \cdot (1 - \hat{p}_{d,oob}) + 2(\hat{p}_{d,oob} - \hat{p}_{d,in})^2 \tag{3}$$

$$PG_{oob}^{(0.5,1)} = \frac{1}{2} \cdot \sum_{d=1}^{D} \hat{p}_{d,oob} \cdot (1 - \hat{p}_{d,oob}) + \hat{p}_{d,in} \cdot (1 - \hat{p}_{d,in}) + (\hat{p}_{d,oob} - \hat{p}_{d,in})^2 \tag{4}$$

$$\widehat{PG}_{oob}^{(1,0)} = \frac{N}{N-1} \cdot \sum_{d=1}^{D} \hat{p}_{d,oob} \cdot (1 - \hat{p}_{d,oob}) \tag{5}$$

$$\tag{6}$$

Our main contributions are to show that

- $PG_{oob}^{(1,2)}$ is equivalent to the *MDI-oob* measure defined in Li et al. (2019a).
- $PG_{oob}^{(1,2)}$ has close connections to the *conditional feature contributions* (CFCs) defined in (Saabas, 2019b),
- Similarly to MDI, both the CFCs as well as the related *SHapley Additive exPlanation* (SHAP) values defined in (Lundberg et al., 2020) are susceptible to “overfitting” to the training data.

- $PG_{oob}^{(0.5,1)}$ is equivalent to the *unbiased split-improvement* measure defined in Zhou and Hooker (2019).

We refer the reader to (Loecher, 2020) for a proof that $\widehat{PG}_{oob}^{(1,0)}$ and $PG_{oob}^{(0.5,1)}$ are unbiased estimators of feature importance in the case of non-informative variables.

3 Conditional feature contributions (CFCs)

The conventional wisdom of estimating the impact of a feature in tree based models is to measure the **node-wise reduction of a loss function**, such as the variance of the output Y, and compute a weighted average of all nodes over all trees for that feature. By its definition, such a *mean decrease in impurity* (MDI) serves only as a global measure and is typically not used to explain a *per-observation, local impact.* Saabas (2019b) proposed the novel idea of explaining a prediction by following the decision path and attributing changes in the expected output of the model to each feature along the path. Figure 1 illustrates the main idea of decomposing each prediction through the sequence of regions that correspond to each node in the tree. Each decision either adds or subtracts from the value given in the parent node and can be attributed to the feature at the node. So, each individual prediction can be defined as the global mean plus the sum of the K feature contributions:

$$f_{pred}(x_i) = \bar{Y} + \sum_{k=1}^{K} f_{T,k}(x_i) \tag{7}$$

where $f_{T,k}(x_i)$ is the contribution from the k-th feature (for tree T), written as a sum over all the inner nodes t such that $v(t) = k$ (Li et al., 2019a)[1]:

$$f_{T,k}(x_i) = \sum_{t \in I(T): v(t)=k} \left[\mu_n\left(t^{left}\right) \mathbb{1}\left(x_i \in R_{t^{left}}\right) + \mu_n\left(t^{right}\right) \mathbb{1}\left(x_i \in R_{t^{right}}\right) - \mu_n(t) \mathbb{1}\left(x_{\in} R_t\right) \right] \tag{8}$$

where $v(t)$ is the feature chosen for the split at node t.

[1] Appendix 8.1 contains expanded definitions and more thorough notation.

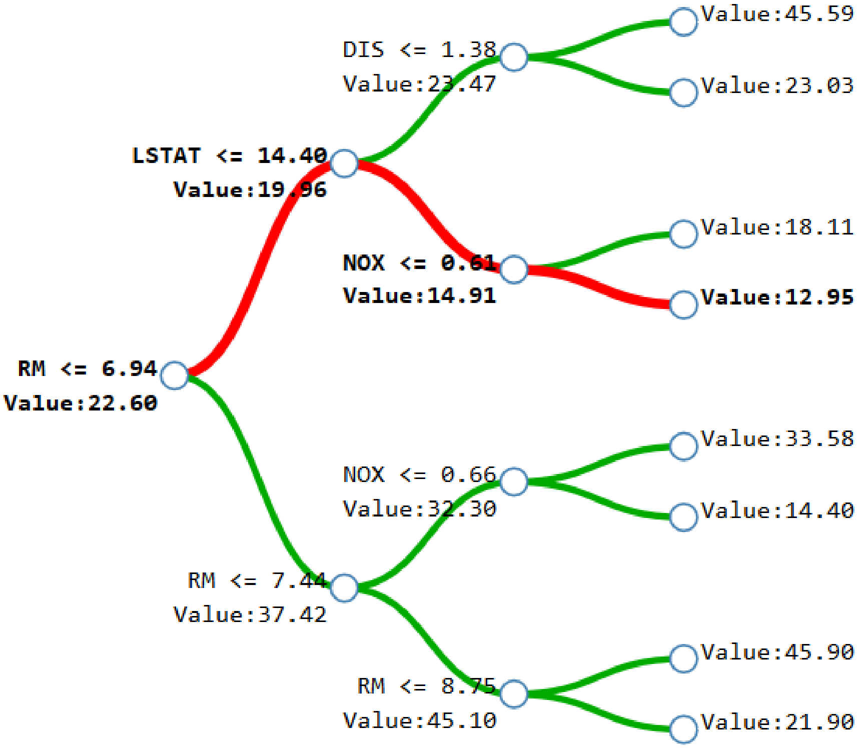

Figure 1: Taken from (Saabas, 2019a): Depicted is a regression decision tree to predict housing prices. The tree has conditions on each internal node and a value associated with each leaf (i.e. the value to be predicted). But additionally, the value at each internal node i.e. the mean of the response variables in that region, is shown. The red path depicts an example prediction $Y = 12.95$, broken down as follows:

$\mathbf{12.95} \approx \mathbf{22.60}(\bar{Y}) - \mathbf{2.64}(\text{loss from RM}) - \mathbf{5.04}(\text{loss from LSTAT}) - \mathbf{1.96}\ (\text{loss from NOX})$

A "local" feature importance score can be obtained by summing Eq. (8) over all trees. Adding these local explanations over all data points yields a "global" importance score:

$$Imp_{global}(k) = \sum_{i=1}^{N} |Imp_{local}(k, x_i)| = \sum_{i=1}^{N} \frac{1}{n_T} \sum_{T} |f_{T,k}(x_i)| \tag{9}$$

In the light of wanting to explain the predictions from tree based machine learning models, the "Saabas algorithm" is extremely appealing, because

- The positive and negative contributions from nodes convey directional information unlike the strictly positive purity gains.
- By combining many local explanations we can represent global structure while retaining local faithfulness to the original model.
- The expected value of every node in the tree can be estimated efficiently by averaging the model output over all the training samples that pass through that node.
- The algorithm has been implemented and is easily accessible in a python (Saabas, 2019b) and R (Sun, 2020) library.

However, Lundberg et al. (2020) pointed out that it is strongly biased to alter the impact of features based on their distance from the root of a tree. This causes Saabas values to be inconsistent, which means one can modify a model to make a feature clearly more important, and yet the Saabas value attributed to that feature will decrease. As a solution, the authors developed an algorithm ("TreeExplainer") that computes local explanations based on exact Shapley values in polynomial time. This provides local explanations with theoretical guarantees of local accuracy and consistency. A python library is available at `https://github.com/slundberg/shap`. One should not forget though that the same idea of adding *conditional feature contributions* lies at the heart of *TreeExplainer*.

In this section, we call attention to another source of bias which is the result of using the same (inbag) data to (i) greedily split the nodes during the growth of the tree and (ii) computing the node-wise changes in prediction. We use the well known titanic data set to

illustrate the perils of putting too much faith into importance scores which are based entirely on training data - not on OOB samples - and make no attempt to discount node splits in deep trees that are spurious and will not survive in a validation set.
In the following model[2] we include *passengerID* as a feature along with the more reasonable *Age*, *Sex* and *Pclass*.

Figure 2 below depicts the distribution of the individual, “local” feature contributions, preserving their sign. The large variations for the variables with high cardinality (*Age, passengerID*) are worrisome. We know that the impact of *Age* was modest and that *passengerID* has no impact on survival but when we sum the absolute values, both features receive sizable importance scores, as shown in Figure 3. This troubling result is robust to random shuffling of the ID. Section 5 will point out a close analogy between the well known MDI score and the more recent measure based on the conditional feature contributions.

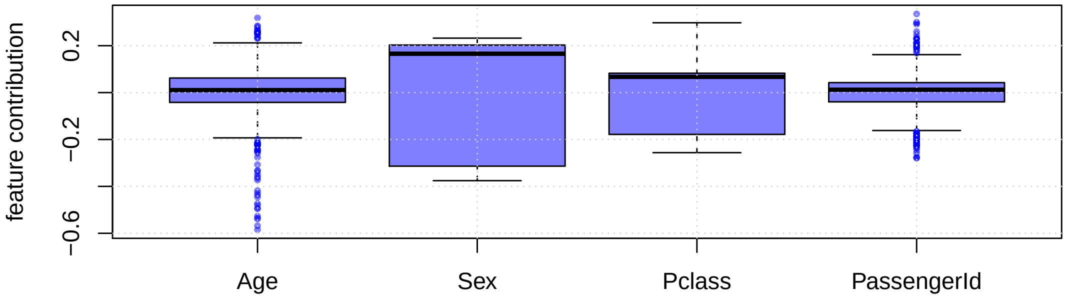

Figure 2: Conditional feature contributions (TreeInterpreter) for the Titanic data.

4 SHAP values

Lundberg et al. (2020) introduce a new local feature attribution method for trees based on

[2] In all random forest simulations, we choose $mtry = 2, ntrees = 100$ and exclude rows with missing *Age*

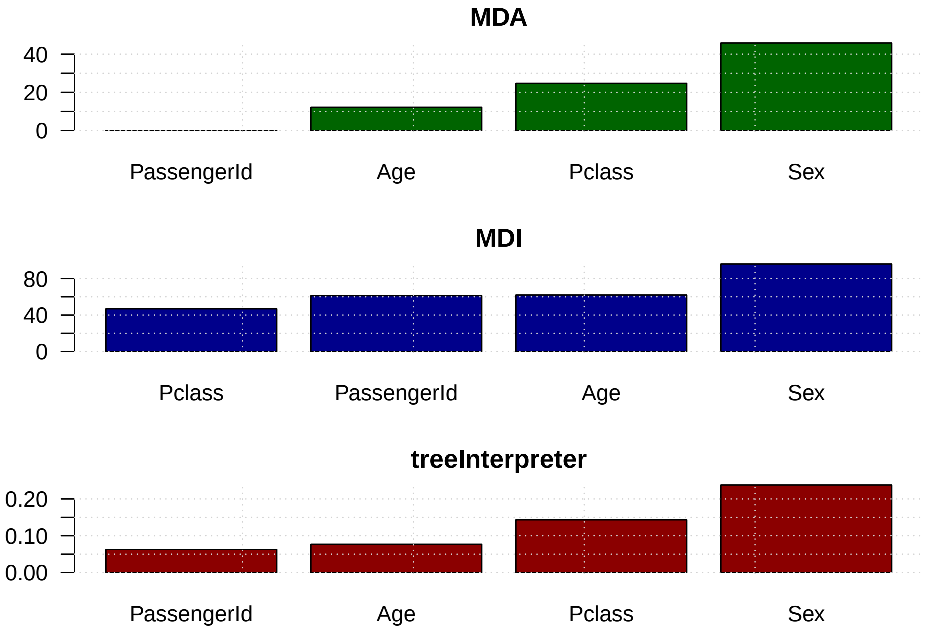

Figure 3: Permutation importance (MDA, top panel) versus Mean decrease impurity (MDI, left panel) versus conditional feature contributions (TreeInterpreter) for the Titanic data. The permutation based importance (MDA) is not fooled by the irrelevant ID feature. This is maybe not unexpected as the IDs should bear no predictive power for the out-of-bag samples.

SHapley Additive exPlanation (SHAP) values which fall in the class of *additive feature attribution methods*. The authors point to results from game theory implying that Shapley values are the only way to satisfy three important properties: *local accuracy, consistency, and missingness*. In this section we show that even (SHAP) values suffer from (i) a strong dependence on feature cardinality, and (ii) assign non zero importance scores to uninformative features, which would violate the *missingness* property. We begin by extending the Titanic example from the previous section and find that *TreeExplainer* also assigns a non

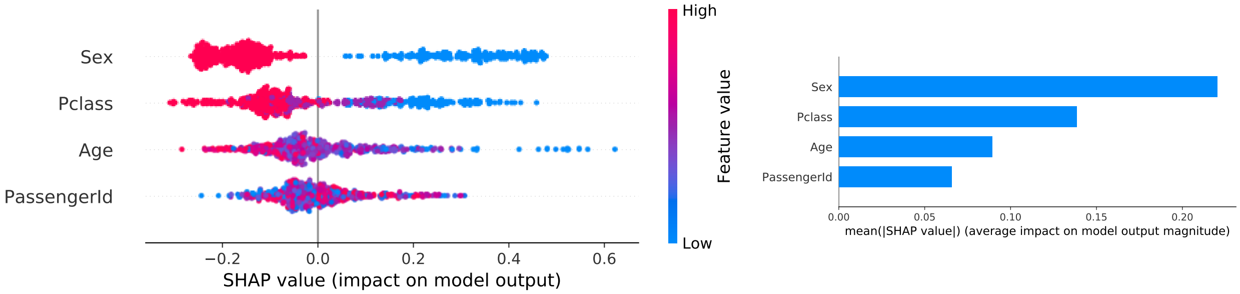

Figure 4: SHapley Additive exPlanation (SHAP) values (TreeExplainer) for the Titanic data.

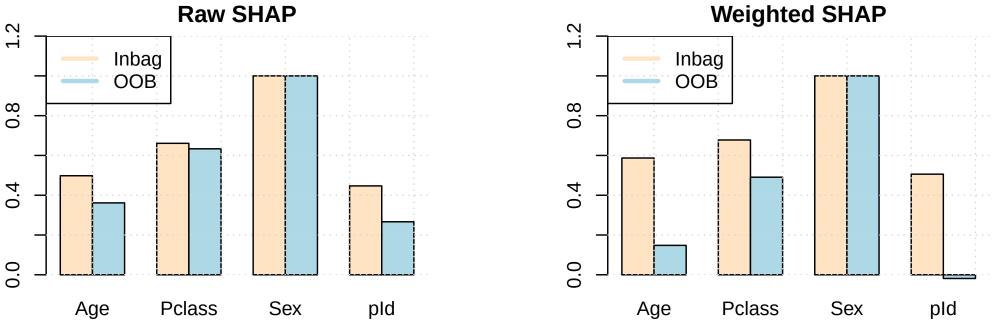

Figure 5: Left graph: "raw" SHAP values for the Titanic data, separately computed for inbag and OOB. Right graph: weighted SHAP values are multiplied by y_i before averaging which eliminates the spurious contributions due to *passengerID* for OOB. Note that we scaled the SHAP values to their respective maxima for easier comparison. (*pId* is short for *passengerID*)

zero value of feature importance to *passengerID*, as shown in Figure 4, which is due to mixing inbag and out-of-bag data for the evaluation. Simply separating the inbag from the OOB SHAP values is not a remedy as shown in the left graph of Figure 5. However, inspired by Li et al. (2019a) (see section 5), we compute weighted SHAP values by mutliplying with y_i before averaging. The right graph of Figure 5 illustrates the elimination of the spurious contributions due to *passengerID* for the OOB SHAP values. Further support for the claims above is given by the following two kinds of simulations.

4.1 Null/Power Simulations

We replicate the simulation design used by Strobl et al. (2007a) where 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, 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 = 120. In the first *null case* all predictor variables and the response are sampled independently. We would hope that a reasonable variable importance measure would not prefer any one predictor variable over any other. In the second simulation study, the so-called *power case*, the distribution of the response is a binomial process with probabilities that depend on the value of x_2, namely $P(y = 1|X_2 = 1) = 0.35, P(y = 1|X_2 = 2) = 0.65$.

As is evident in the two leftmost panels of Figure 6, both the Gini importance (MDI) and the SHAP values show a strong preference for variables with many categories and the continuous variable. This bias is of course well-known for MDI but maybe unexpected for the SHAP scores which clearly violate the *missingness* property. Encouragingly, both $PG_{OOB}^{(0.5,1)}$ and AIR (Nembrini et al., 2018) yield low scores for all predictors. The notable differences in the variance of the distributions for predictor variables with different scale of measurement or number of categories are unfortunate but to be expected. (The larger the numbers of categories in a multinomial variable, the fewer the numbers of observations per category and the larger therefore the variability of the measured qualities of the splits performed using

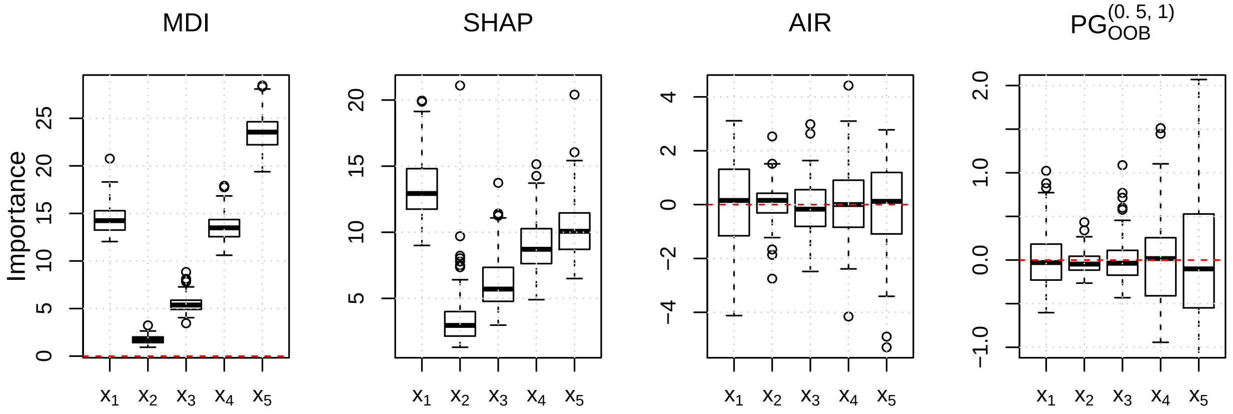

Figure 6: Results of the null case, where none of the predictor variables is informative.

the multinomial variable) The results from the power study are summarized in Figure 7. MDI and SHAP again show a strong bias towards variables with many categories and the continuous variable. At the chosen sigal-to-noise ratio MDI 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. Both $PG_{OOB}^{(0.5,1)}$ and AIR clearly succeed in

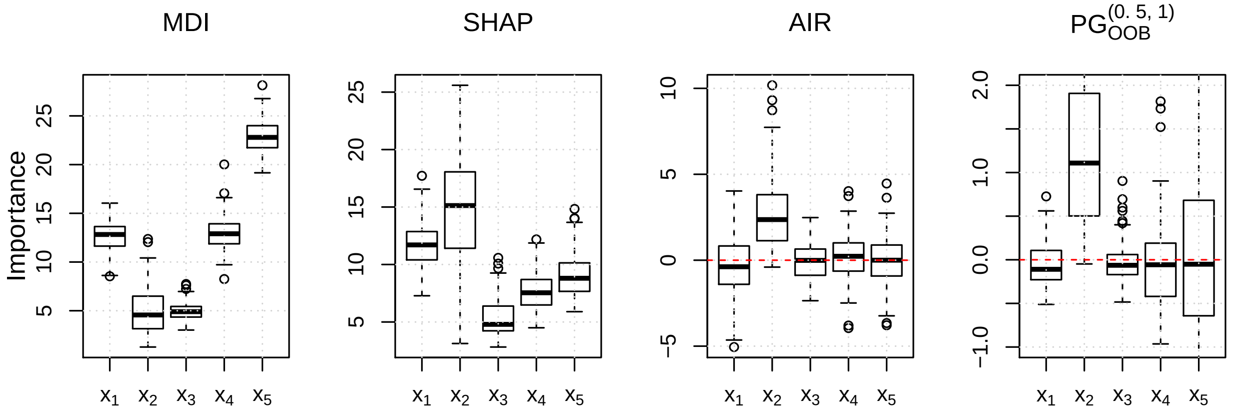

Figure 7: Results of the power study, where only X_2 is informative. Other simulation details as in Fig. 6.

identifying X_2 as the most relevant feature. The large fluctuations of the importance scores for X_4 and especially X_5 are bound to yield moderate "false positive" rates and incorrect rankings in single trials. The signal-to-noise separation for the SHAP values is moderate but

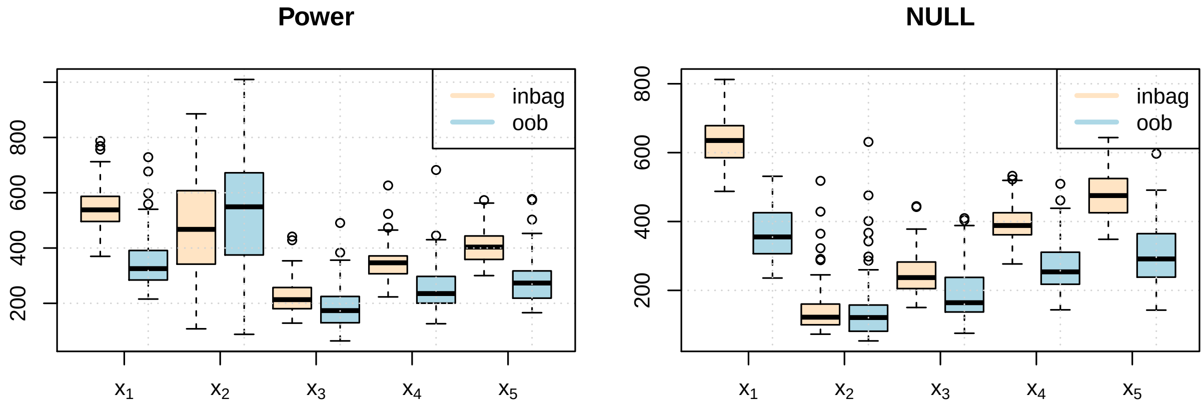

Figure 8: Weighted SHAP values, as explined in the text. Left graph: power study, where only X_2 is informative. Right graph: null case, where none of the predictor variables is informative. Other simulation details as in Figs. 6, 7

can be greatly improved by mutliplying with y_i before averaging (in analogy to Figure 5 and section 5), as shown in Fig. 8.

4.2 Noisy feature identification

For a more systematic comparison of the 4 proposed penalized Gini scores, we closely follow the simulations outlined in Li et al. (2019b) involving discrete features with different number of distinct values, which poses a critical challenge for MDI. The data has 1000 samples with 50 features. All features are discrete, with the jth feature containing $j+1$ distinct values $0, 1, \ldots, j$. We randomly select a set S of 5 features from the first ten as relevant features. The remaining features are noisy features. All samples are i.i.d. and all features are independent. We generate the outcomes using the following rule:

$$P(Y=1|X) = \text{Logistic}(\frac{2}{5}\sum_{j\in S} x_j/j - 1)$$

Treating the noisy features as label 0 and the relevant features as label 1, we can evaluate a feature importance measure in terms of its area under the receiver operating characteristic curve (AUC). We grow 100 deep trees (minimum leaf size equals 1, $m_{try} = 3$), repeat the whole process 100 times and report the average AUC scores for each method in Table 1. For this simulated setting, $\widehat{PG}_{oob}^{(0.5,1)}$ achieves the best AUC score under all cases, most likely because of the separation of the signal from noise mentioned above. We notice that the AUC score for the OOB-only $\widehat{PG}_{oob}^{(1,0)}$ is competitive to the permutation importance, SHAP and the AIR score.

$\widehat{PG}_{oob}^{(1,0)}$	$PG_{oob}^{(1,0)}$	$\widehat{PG}_{oob}^{(0.5,1)}$	$PG_{oob}^{(0.5,1)}$	SHAP	SHAP_{in}	SHAP_{oob}	AIR	MDA	MDI
0.66	0.28	0.92	0.78	0.66	0.56	0.73	0.68	0.65	0.10

Table 1: Average AUC scores for noisy feature identification. *MDA* = permutation importance, *MDI* = (default) Gini impurity. The $\widehat{PG}_{oob}$ scores apply the variance bias correction $n/(n-1)$. The SHAP_{in}, SHAP_{oob} scores are based upon separating the inbag from the oob data.

5 MDI versus CFCs

As elegantly demonstrated by Li et al. (2019a), the MDI of feature k in a tree T can be written as

$$MDI = \frac{1}{|\mathcal{D}^{(T)}|} \sum_{i \in \mathcal{D}^{(T)}} f_{T,k}(x_i) \cdot y_i \tag{10}$$

where $\mathcal{D}^{(T)}$ is the bootstrapped or subsampled data set of the original data $\mathcal{D}$. Since $\sum_{i \in \mathcal{D}^{(T)}} f_{T,k}(x_i) = 0$, we can view MDI essentially as the sample covariance between $f_{T,k}(x_i)$ and y_i on the bootstrapped dataset $\mathcal{D}^{(T)}$. Alternatively, we can view MDI as a particular weighted average of the CFCs, which for the special case of binary classification ($y_i \in \{0, 1\}$) means that one only adds up those CFCs for which $y_i = 1$.

$$MDI = \frac{1}{|\mathcal{D}^{(T)}[y_i = 1]|} \sum_{i \in \mathcal{D}^{(T)}[y_i=1]} f_{T,k}(x_i) \tag{11}$$

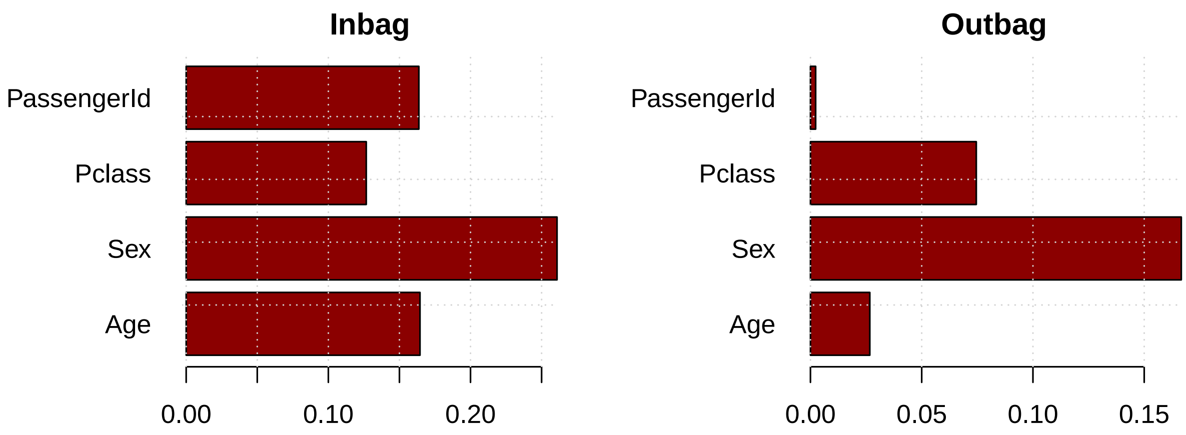

Figure 9: MDI as a restricted sum of conditional feature contributions defined by Eq. (11) for the (left panel) inbag and (right panel) outbag data, respectively.

Figure 9 shows this new measure separately for the inbag and outbag components; the middle panel of Figure 3 is proportional to the left panel. The misleadingly high contribution of *passengerID* is due to the well known shortcoming of MDI: RFs use the training data $\mathcal{D}^{(T)}$ to construct the functions $f_{T,k}()$, then MDI uses the same data to evaluate (10).

6 Debiasing MDI via oob samples

In this section we give a short version of the proof that $PG_{oob}^{(1,2)}$ is equivalent to the *MDI-oob* measure defined in Li et al. (2019a). For clarity we assume binary classification; Appendix 6 contains an expanded version of the proof including the multi-class case. *MDI-oob* is based on the usual variance reduction per node as shown in Eq. (34) (proof of Proposition (1)), but with a "variance" defined as the mean squared deviations of y_{oob} from the inbag mean μ_{in}:

$$\Delta_I(t) = \frac{1}{N_n(t)} \cdot \sum_{i \in D(T)} (y_{i,oob} - \mu_{n,in})^2 1(x_i \in R_t) - \ldots$$

We can, of course, rewrite the variance as

$$\frac{1}{N_n(t)} \cdot \sum_{i \in D(T)} (y_{i,oob} - \mu_{n,in})^2 = \frac{1}{N_n(t)} \cdot \sum_{i \in D(T)} (y_{i,oob} - \mu_{n,oob})^2 + (\mu_{n,in} - \mu_{n,oob})^2 \tag{12}$$

$$= p_{oob} \cdot (1 - p_{oob}) + (p_{in} - p_{oob})^2 \tag{13}$$

where the last equality is for Bernoulli y_i, in which case the means $\mu_{in/oob}$ become proportions $p_{in/oob}$ and the first sum is equal to the binomial variance $p_{oob} \cdot (1 - p_{oob})$. The final expression is effectively equal to $PG_{oob}^{(1,2)}$.

Lastly, we now show that $PG_{oob}^{(0.5,1)}$ is equivalent to the *unbiased split-improvement* measure defined in Zhou and Hooker (2019). For the binary classificaton case, we can rewrite $PG_{oob}^{(0.5,1)}$ as follows:

$$PG_{oob}^{(0.5,1)} = \frac{1}{2} \cdot \sum_{d=1}^{D} \hat{p}_{d,oob} \cdot (1 - \hat{p}_{d,oob}) + \hat{p}_{d,in} \cdot (1 - \hat{p}_{d,in}) + (\hat{p}_{d,oob} - \hat{p}_{d,in})^2 \tag{14}$$

$$= \hat{p}_{oob} \cdot (1 - \hat{p}_{oob}) + \hat{p}_{in} \cdot (1 - \hat{p}_{in}) + (\hat{p}_{oob} - \hat{p}_{in})^2 \tag{15}$$

$$= \hat{p}_{oob} - \hat{p}_{oob}^2 + \hat{p}_{in} - \hat{p}_{in}^2 + \hat{p}_{oob}^2 - 2\hat{p}_{oob} \cdot \hat{p}_{in} + \hat{p}_{in}^2 \tag{16}$$

$$= \hat{p}_{oob} + \hat{p}_{in} - 2\hat{p}_{oob} \cdot \hat{p}_{in} \tag{17}$$

7 Discussion

Random forests and gradient boosted trees are among the most popular and powerful (Olson et al., 2017) non-linear predictive models used in a wide variety of fields. Lundberg et al. (2020) demonstrate that tree-based models can be more accurate than neural networks and even more interpretable than linear models. In the comprehensive overview of variable importance in regression models Grömping (2015) distinguishes between methods based on (i) variance decomposition and (ii) standardized coefficient sizes, which is somewhat analogous to the difference between (i) MDI and (ii) CFCs. The latter measure the directional impact of $x_{k,i}$ on the outcome y_i, whereas MDI based scores measure a kind of *partial* R_k^2 (if one stretched the analogy to linear models). Li et al. (2019a) ingeniously illustrate the connection between these seemingly fundamentally different methods via eq. (10). And the brilliant extension of CFCs to their Shapley equivalents by Lundberg et al. (2020) bears affinity to the game-theory-based metrics LMG and PMVD (Grömping (2015) and references therein), which are based on averaging the sequential R_k^2 over all orderings of regressors.

In this paper we have (i) connected the proposals to reduce the well known bias in MDI by mixing inbag and oob data (Li et al., 2019a, Zhou and Hooker, 2019) to a common framework (Loecher, 2020), and (ii) pointed out that similar ideas would benefit/debias the *conditional feature contributions* (CFCs) (Saabas, 2019b) as well as the related *SHapley Additive exPlanation* (SHAP) values (Lundberg et al., 2020).

While the main findings are applicable to any tree based method, they are most relevant to random forests (RFs) since (i) oob data are readily available and (ii) RFs typically grow deep trees. Li et al. (2019a) showed a strong dependence of the MDI bias on the depth of the tree: splits in nodes closer to the roots are much more stable and supported by larger sample sizes and hence hardly susceptible to bias. RFs "get away" with the individual overfitting of deep trees to the training data by **averaging** many (hundreds) of separately grown deep trees and often achieve a favorable balance in the bias-variance tradeoff. One reason is certainly that the noisy predictions from individual trees "average out", which is not the case for the summing/averaging of the strictly positive MDI leading to what could be called *interpretational overfitting.* The big advantage of conditional feature contributions is that positive and negative contributions can cancel across trees making it less prone to that type of overfitting. However, we have provided evidence that both the CFCs as well as the SHAP values are still susceptible to "overfitting" to the training data and can benefit from evaluation on oob data.

References

L. Breiman. Random forests. *Machine Learning*, 45, 2001. 10.1023/A:1010933404324. URL `https://doi.org/10.1023/A:1010933404324`.

Ramón Díaz-Uriarte and Sara Alvarez De Andres. Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7(1):3, 2006.

Ulrike Grömping. Variable importance assessment in regression: linear regression versus random forest. *The American Statistician*, 63(4):308–319, 2009.

Ulrike Grömping. Variable importance in regression models. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(2):137–152, 2015.

Torsten Hothorn, Kurt Hornik, and Achim Zeileis. Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical statistics*, 15(3): 651–674, 2006.

Hyunjoong Kim and Wei-Yin Loh. Classification trees with unbiased multiway splits. *Journal of the American Statistical Association*, 96(454):589–604, 2001.

Xiao Li, Yu Wang, Sumanta Basu, Karl Kumbier, and Bin Yu. A debiased mdi feature importance measure for random forests. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8049–8059. Curran Associates, Inc., 2019a. URL `http://papers.nips.cc/paper/9017-a-debiased-mdi-feature-importance-measure-for-random-forests`.

Xiao Li, Yu Wang, Sumanta Basu, Karl Kumbier, and Bin Yu. A debiased mdi feature importance measure for random forests. *arXiv preprint arXiv:1906.10845*, 2019b.

Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R News*, 2 (3):18–22, 2002. URL `https://CRAN.R-project.org/doc/Rnews/`.

Markus Loecher. Unbiased variable importance for random forests. *Communications in Statistics - Theory and Methods*, 0(0):1–13, 2020. 10.1080/03610926.2020.1764042. URL `https://doi.org/10.1080/03610926.2020.1764042`.

Wei-Yin Loh and Yu-Shan Shih. Split selection methods for classification trees. *Statistica sinica*, pages 815–840, 1997.

Scott M Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable ai for trees. *Nature machine intelligence*, 2(1): 56–67, 2020.

Bjoern H Menze, B Michael Kelm, Ralf Masuch, Uwe Himmelreich, Peter Bachert, Wolfgang Petrich, and Fred A Hamprecht. A comparison of random forest and its gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC Bioinformatics*, 10(1):213, 2009.

Stefano Nembrini, Inke R König, and Marvin N Wright. The revival of the gini importance? *Bioinformatics*, 34(21):3711–3718, 2018.

Randal S Olson, William La Cava, Zairah Mustahsan, Akshay Varik, and Jason H Moore. Data-driven advice for applying machine learning to bioinformatics problems. *arXiv preprint arXiv:1708.05070*, 2017.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct): 2825–2830, 2011.

Ando Saabas. Interpreting random forests, 2019a. URL `http://blog.datadive.net/interpreting-random-forests/`.

Ando Saabas. Treeinterpreter library, 2019b. URL `https://github.com/andosa/treeinterpreter`.

Marco Sandri and Paola Zuccolotto. A bias correction algorithm for the gini variable importance measure in classification trees. *Journal of Computational and Graphical Statistics*, 17(3):611–628, 2008.

Y-S Shih. A note on split selection bias in classification trees. *Computational statistics & data analysis*, 45(3):457–466, 2004.

Yu-Shan Shih and Hsin-Wen Tsai. Variable selection bias in regression trees with constant fits. *Computational statistics & data analysis*, 45(3):595–607, 2004.

C. Strobl, A. L. Boulesteix, A. Zeileis, and T. Hothorn. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics*, 8, 2007a. 10.1186/1471-2105-8-25. URL `https://doi.org/10.1186/1471-2105-8-25`.

Carolin Strobl, Anne-Laure Boulesteix, and Thomas Augustin. Unbiased split selection for classification trees based on the gini index. *Computational Statistics & Data Analysis*, 52 (1):483–501, 2007b.

Qingyao Sun. *tree.interpreter: Random Forest Prediction Decomposition and Feature Importance Measure*, 2020. URL `https://CRAN.R-project.org/package=tree.interpreter`. R package version 0.1.1.

Marvin N Wright and Andreas Ziegler. ranger: A fast implementation of random forests for high dimensional data in c++ and r. *arXiv preprint arXiv:1508.04409*, 2015.

Marvin N Wright, Stefan Wager, Philipp Probst, and Maintainer Marvin N Wright. Package ranger. 2017.

Zhengze Zhou and Giles Hooker. Unbiased measurement of feature importance in tree-based methods. *arXiv preprint arXiv:1903.05179*, 2019.

8 Appendix

8.1 Background and notations

Definitions needed to understand Eq. (8). (The following paragraph closely follows the definitions in Li et al. (2019a).)

Random Forests (RFs) are an ensemble of classification and regression trees, where each tree T defines a mapping from the feature space to the response. Trees are constructed independently of one another on a bootstrapped or subsampled data set $\mathcal{D}^{(T)}$ of the original data $\mathcal{D}$. Any node t in a tree T represents a subset (usually a hyper-rectangle) R_t of the feature space. A split of the node t is a pair (k, z) which divides the hyper-rectangle R_t into two hyper-rectangles $R_t \cap \mathbb{1}\left(X_k \leq z\right)$ and $R_t \cap \mathbb{1}\left(X_k > z\right)$ corresponding to the left child t left and right child t right of node t, respectively. For a node t in a tree $T, N_n(t) = \left|\left\{i \in \mathcal{D}^{(T)} : \mathbf{x}_i \in R_t\right\}\right|$ denotes the number of samples falling into R_t and

$$\mu_n(t) := \frac{1}{N_n(t)} \sum_{i:\mathbf{x}_i \in R_t} y_i$$

We define the set of inner nodes of a tree T as $I(T)$.

8.2 Variance Reduction View

Here, we provide a full version of the proof sketched in section 6 which leans heavily on the proof of Proposition (1) in Li et al. (2019b) .

We consider the usual variance reduction per node but with a "variance" defined as the mean squared deviations of y_{oob} from the inbag mean μ_{in}:

$$\begin{aligned} \Delta_{\mathcal{I}}(t) = \frac{1}{N_n(t)} \sum_{i \in \mathcal{D}^{(T)}} & \left[y_{i,oob} - \mu_{n,in}(t)\right]^2 \mathbb{1}\left(\mathbf{x}_i \in R_t\right) \\ & - \left[y_{i,oob} - \mu_{n,in}\left(t^{\text{left}}\right)\right]^2 \mathbb{1}\left(\mathbf{x}_i \in R_{t^{\text{left}}}\right) - \left[y_{i,oob} - \mu_{n,in}\left(t^{\text{right}}\right)\right]^2 \mathbb{1}\left(\mathbf{x}_i \in R_{\text{right}}\right) \end{aligned} \tag{18}$$

$$
\begin{aligned}
= \frac{1}{N_n(t)} \sum_{i \in \mathcal{D}^{(T)}} & \left([y_{i,oob} - \mu_{n,oob}(t)]^2 + [\mu_{n,in}(t) - \mu_{n,oob}(t)]^2 \right) \mathbb{1}\left(\mathbf{x}_i \in R_t\right) \\
& - \left(\left[y_{i,oob} - \mu_{n,oob}(t^{\text{left}})\right]^2 + \left[\mu_{n,in}(t^{\text{left}}) - \mu_{n,oob}(t^{\text{left}})\right]^2 \right) \mathbb{1}\left(\mathbf{x}_i \in R_{t^{\text{left}}}\right) \\
& - \left(\left[y_{i,oob} - \mu_{n,oob}(t^{\text{right}})\right]^2 + \left[\mu_{n,in}(t^{\text{right}}) - \mu_{n,oob}(t^{\text{right}})\right]^2 \right) \mathbb{1}\left(\mathbf{x}_i \in R_{\text{right}}\right)
\end{aligned}
\tag{19}
$$

$$
\begin{aligned}
= & \frac{1}{N_n(t)} \underbrace{\sum_{i \in \mathcal{D}^{(T)}} \left\{ [y_{i,oob} - \mu_{n,oob}(t)]^2 \, \mathbb{1}\left(\mathbf{x}_i \in R_t\right) \right\}}_{N_n(t) \cdot p_{oob}(t) \cdot (1 - p_{oob}(t))} + \underbrace{[\mu_{n,in}(t) - \mu_{n,oob}(t)]^2}_{[p_{oob}(t) - p_{in}(t)]^2} \\
- & \frac{1}{N_n(t)} \underbrace{\sum_{i \in \mathcal{D}^{(T)}} \left\{ \left[y_{i,oob} - \mu_{n,oob}(t^{\text{left}})\right]^2 \mathbb{1}\left(\mathbf{x}_i \in R_{t^{\text{left}}}\right) \right\}}_{N_n(t^{\text{left}}) \cdot p_{oob}(t^{\text{left}}) \cdot (1 - p_{oob}(t^{\text{left}}))} + \underbrace{\left[\mu_{n,in}(t^{\text{left}}) - \mu_{n,oob}(t^{\text{left}})\right]^2}_{\left[p_{oob}(t^{\text{left}}) - p_{in}(t^{\text{left}})\right]^2} \\
- & \frac{1}{N_n(t)} \underbrace{\sum_{i \in \mathcal{D}^{(T)}} \left\{ \left[y_{i,oob} - \mu_{n,oob}(t^{\text{right}})\right]^2 \mathbb{1}\left(\mathbf{x}_i \in R_{\text{right}}\right) \right\}}_{N_n(t^{\text{right}}) \cdot p_{oob}(t^{\text{right}}) \cdot (1 - p_{oob}(t^{\text{right}}))} + \underbrace{\left[\mu_{n,in}(t^{\text{right}}) - \mu_{n,oob}(t^{\text{right}})\right]^2}_{\left[p_{oob}(t^{\text{right}}) - p_{in}(t^{\text{right}})\right]^2}
\end{aligned}
$$

where the last equality is for Bernoulli y_i, in which case the means $\mu_{in/oob}$ become proportions $p_{in/oob}$ and we replace the squared deviations with the binomial variance $p_{oob} \cdot (1 - p_{oob})$. The final expression is then

$$
\begin{aligned}
\Delta_{\mathcal{I}}(t) = & \, p_{oob}(t) \cdot (1 - p_{oob}(t)) + [p_{oob}(t) - p_{in}(t)]^2 \\
& - \frac{N_n(t^{\text{left}})}{N_n(t)} \left(p_{oob}(t^{\text{left}}) \cdot \left(1 - p_{oob}(t^{\text{left}})\right) + \left[p_{oob}(t^{\text{left}}) - p_{in}(t^{\text{left}})\right]^2 \right) \\
& - \frac{N_n(t^{\text{right}})}{N_n(t)} \left(p_{oob}(t^{\text{right}}) \cdot \left(1 - p_{oob}(t^{\text{right}})\right) + \left[p_{oob}(t^{\text{right}}) - p_{in}(t^{\text{right}})\right]^2 \right)
\end{aligned}
\tag{20}
$$

which, of course, is exactly the impurity reduction due to $PG_{oob}^{(1,2)}$:

$$
\Delta_{\mathcal{I}}(t) = PG_{oob}^{(1,2)}(t) - \frac{N_n(t^{\text{left}})}{N_n(t)} PG_{oob}^{(1,2)}(t^{\text{left}}) - \frac{N_n(t^{\text{right}})}{N_n(t)} PG_{oob}^{(1,2)}(t^{\text{right}})
\tag{21}
$$

Another, somewhat surprising view of MDI is given by Eqs. (10) and (8), which for binary classification reads as:

$$\begin{aligned} MDI &= \frac{1}{|\mathcal{D}^{(T)}|} \sum_{t \in I(T):v(t)=k} \sum_{i \in \mathcal{D}^{(T)}} \left[\mu_n\left(t^{left}\right) \mathbb{1}\left(x_i \in R_{t^{left}}\right) + \mu_n\left(t^{right}\right) \mathbb{1}\left(x_i \in R_{t^{right}}\right) - \mu_n(t) \mathbb{1}\left(x_{\in} R_t\right)\right] \cdot y_i \\ &= \frac{1}{|\mathcal{D}^{(T)}|} \sum_{t \in I(T):v(t)=k} -p_{in}(t)^2 + \frac{N_n(t^{\text{left}})}{N_n(t)} p_{in}(t^{\text{left}})^2 + \frac{N_n(t^{\text{right}})}{N_n(t)} p_{in}(t^{\text{right}})^2 \end{aligned} \tag{22}$$

and for the oob version:

$$MDI_{oob} = -p_{in}(t) \cdot p_{oob}(t) + \frac{N_n(t^{\text{left}})}{N_n(t)} p_{in}(t^{\text{left}}) \cdot p_{oob}(t^{\text{left}}) + \frac{N_n(t^{\text{right}})}{N_n(t)} p_{in}(t^{\text{right}}) \cdot p_{oob}(t^{\text{right}}) \tag{23}$$

The above expressions suggest that node impurity could be simply measured by $-p_{in}(t)^2$, and $-p_{in}(t) \cdot p_{oob}(t)$, respectively. While this would be

8.3 $\mathbf{E}(\mathbf{\Delta}\widehat{\mathbf{PG}}^{(\mathbf{0})}_{\mathbf{oob}}) = \mathbf{0}$

The decrease in impurity (ΔG) for a parent node m is the weighted difference between the Gini importance[3] $G(m) = \hat{p}_m(1 - \hat{p}_m)$ and those of its left and right children:

$$\Delta G(m) = G(m) - \left[N_{m_l} G(m_l) - N_{m_r} G(m_r)\right] / N_m$$

We assume that the node m splits on an **uninformative** variable X_j, i.e. X_j and Y are independent.

We will use the short notation $\sigma^2_{m,.} \equiv p_{m,.}(1 - p_{m,.})$ for . either equal to *oob* or *in* and rely on the following facts and notation:

1. $E[\hat{p}_{m,oob}] = p_{m,oob}$ is the "population" proportion of the class label in the OOB test data (of node m).

2. $E[\hat{p}_{m,in}] = p_{m,in}$ is the "population" proportion of the class label in the inbag test data (of node m).

3. $E[\hat{p}_{m,oob}] = E[\hat{p}_{m_l,oob}] = E[\hat{p}_{m_r,oob}] = p_{m,oob}$

[3] For easier notation we have (i) left the multiplier 2 and (ii) omitted an index for the class membership

4. $E[\hat{p}^2_{m,oob}] = var(\hat{p}_{m,oob}) + E[\hat{p}_{m,oob}]^2 = \sigma^2_{m,oob}/N_m + p^2_{m,oob}$
   $\Rightarrow E[G_{oob}(m)] = E[\hat{p}_{m,oob}] - E[\hat{p}^2_{m,oob}] = \sigma^2_{m,oob} \cdot \left(1 - \frac{1}{N_m}\right)$
   $\Rightarrow E[\widehat{G}_{oob}(m)] = \sigma^2_{m,oob}$

5. $E[\hat{p}_{m,oob} \cdot \hat{p}_{m,in}] = E[\hat{p}_{m,oob}] \cdot E[\hat{p}_{m,in}] = p_{m,oob} \cdot p_{m,in}$

Equalities 3 and 5 hold because of the independence of the inbag and out-of-bag data as well as the independence of X_j and Y.

We now show that $\mathbf{E(\Delta PG^{(0)}_{oob}) \neq 0}$ We use the shorter notation $G_{oob} = PG^{(0)}_{oob}$:

$$\begin{aligned}
E[\Delta G_{oob}(m)] &= E[G_{oob}(m)] - \frac{N_{m_l}}{N_m} E[G_{oob}(m_l)] - \frac{N_{m_r}}{N_m} E[G_{oob}(m_r)] \\
&= \sigma^2_{m,oob} \cdot \left[1 - \frac{1}{N_m} - \frac{N_{m_l}}{N_m}\left(1 - \frac{1}{N_{m_l}}\right) - \frac{N_{m_r}}{N_m}\left(1 - \frac{1}{N_{m_r}}\right)\right] \\
&= \sigma^2_{m,oob} \cdot \left[1 - \frac{1}{N_m} - \frac{N_{m_l} + N_{m_r}}{N_m} + \frac{2}{N_m}\right] = \frac{\sigma^2_{m,oob}}{N_m}
\end{aligned}$$

We see that there is a bias if we used only OOB data, which becomes more pronounced for nodes with smaller sample sizes. This is relevant because visualizations of random forests show that the splitting on uninformative variables happens most frequently for "deeper" nodes.

The above bias is due to the well known bias in variance estimation, which can be eliminated with the bias correction (5), as outlined in the main text. We now show that the bias for this modified Gini impurity is zero for OOB data. As before, $\widehat{G}_{oob} = \widehat{PG}^{(0)}_{oob}$:

$$\begin{aligned}
E[\Delta \widehat{PG}_{oob}(m)] &= E[\widehat{G}_{oob}(m)] - \frac{N_{m_l}}{N_m} E[\widehat{G}_{oob}(m_l)] - \frac{N_{m_r}}{N_m} E[\widehat{G}_{oob}(m_r)] \\
&= \sigma^2_{m,oob} \cdot \left[1 - \frac{N_{m_l} + N_{m_r}}{N_m}\right] = 0
\end{aligned}$$