A Comparison of Tree Ensemble Methods: Can we see the perfect tree in the forest?

Put, J.M.M.S. van de

A Comparison of Tree Ensemble Methods

Can we see the perfect tree in the forest?

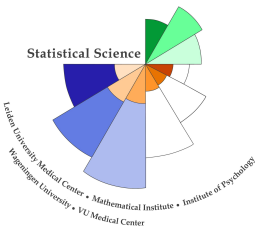
Jeanne M.M.S. van de Put, MSc.

Thesis advisor: E. Dusseldorp, PhD.
Second supervisor: M. Bouts, PhD.
Faculty of Social Sciences, Leiden University

MASTER THESIS

Date: March 20, 2017

Universiteit Leiden



STATISTICAL SCIENCE
FOR THE LIFE AND BEHAVIOURAL SCIENCES

Contents

Abstract

Random forests is generally known as an excellent classifier that is flexible in the types of data it is applied to. Despite this characteristic, it is also regarded as a 'black box' classifier: its ensembles comprise of hundreds of complex tree members. This is a major drawback for certain applications, where insight in the involvement of variables that account for certain outcomes is essential (e.g., medical diagnosis problems for identifying diseased individuals). There are however more recent methods that produce ensembles reduced in size by selecting the most important ensemble members. Some of these methods also yield ensemble members with simple structures to increase interpretation possibilities. Our selection of such methods comprises optimal trees ensemble (OTE), node harvest, and rule ensembles. These methods were assessed through a simulation study and an application to an MRI dataset on Alzheimer's disease classification, to determine predictive performance and information recovery to estimate suitability for interpretational purposes. Random forests was taken as benchmark for predictive performance and baseline for improvement of interpretation. We focussed solely on binary classification.

The benchmark random forests had generally good predictive performances and among the best variable importance recovery. It was still the superior classifier in high-dimensional settings. OTE often had similar predictive and variable importance recovery. It did however not have any advantage over random forests regarding suitability for interpretation. Node harvest had reasonable interaction recovery and good variable split point recovery, albeit at the cost of predictive and variable importance recovery performances. Rule ensembles proved to be a suitable alternative for random forests that produces models suitable for interpretation with comparable or better accuracy, but only when the dataset has clear signal. In noisy or high-dimensional settings, there still is no suitable, more interpretable tree ensemble alternative to random forests amongst the studied methods. Such settings still benefit from ensembles with numerous highly complex trees.

1) Introduction

Decision tree-based methods, or recursive partitioning methods, are supervised learning techniques that use a nonparametric approach to analysing a dataset. Classification And Regression Trees (CART; Breiman et al. 1984) is a widely used decision tree algorithm. Such methods however have one particular disadvantage: they exhibit high variance (Hastie, Tibshirani, and Friedman 2009). This causes these models to generalize poorly and hence make single trees weak predictors. An effect of this high variance is when data with slightly different values are used for tree construction, the resulting tree might be very different from the first one.

There are several decision tree extensions that overcome this variance problem by producing an ensemble of trees. We will concentrate on four of these extensions.

The first one is random forests (Breiman 2001). This algorithm grows many CART trees that collectively predict the outcome. It works in a similar fashion as bootstrap aggregating (bagging; Breiman 1996), another ensemble algorithm, except that extra randomness is introduced in the tree growing process. For every split in the growing process the tree is forced to pick a variable from a random subset of variables. This introduces more diversity in the ensemble that leads to improved accuracy over bagging. Furthermore, correlation between trees is reduced by growing trees independently of each other by drawing new bootstrap samples every iteration. This high forest diversity and low correlation between trees results in higher predictive accuracy and better generalizability of the produced forest ensemble compared to a single tree. As the grown trees remain unpruned and hence quite large, random forests is able to capture underlying (complex) interactions in the data.

Within machine learning, the ensemble method random forests has very good prediction properties. However, unlike for example single decision tree methods, random forests is unsuitable for interpretational purposes, making it an unappealing method for explanatory purposes. Also, despite that usually a few hundred trees are grown in a random forest, a smaller but diverse tree ensemble would suffice from achieving similar accuracy (e.g., Bernard, Heutte, and Adam 2009a; Latinne, Debeir, and Decaestecker 2001).

The second decision tree ensemble-based extension is Optimal Trees Ensembles (OTE; Khan et al. 2016), which tackles mainly the latter issue of reducing ensemble size. It is a recently proposed method aiming to find an optimal subset of trees of a random forest that significantly reduces ensemble size while retaining similar prediction accuracy. The strongest and most diverse trees of a random forest are selected to form an optimal ensemble of trees.

Two other extensions are Node Harvest (Meinshausen 2010) and Rule Ensembles (Friedman and Popescu 2008) that both make ensembles of nodes/rules from decomposed random forest trees and focus more on interpretability. These two methods seek the most important rules involved in predicting outcomes from their initial ensembles, not only producing a small(er) ensemble but also making it possible to give insight into the most important prediction rules and corresponding responses. The main differences between these two methods are the underlying mechanisms to generate trees and the prediction styles: node harvest prediction rules are similar to decision tree prediction rules (based on averaging), while for rule ensembles the predicted value is the result of a summation of rule coefficient values.

These three other methods are extensions of the existing random forest algorithm, aiming to enhance interpretability and reduce final ensemble size. In this study, random forests and these three methods are compared with each other. Random forests can serve as a benchmark for comparison with other methods such as these three other extensions, as it is usually an excellent predictor but suffers from some major disadvantages. The main question of this research is to what extent random forests is interpretable, whether there is a method based on random forests that is even more interpretable and if that possibly is at the expense of accuracy. The aim of this thesis is to seek for a tree ensemble method that has the desirable properties of random forests with the added value of enhanced interpretation.

For this purpose, random forests and related methods are studied in depth to compare predictive performances, ensemble reduction, to what extent these methods are interpretable and if ensemble size/interpretability is at the expense of accuracy. This is done through a demonstration on a toy dataset, a large simulation and an application to a real dataset. The focus will lie on performances in binary classification settings only. In every part of the study, random forests is taken as benchmark method to which the other ensemble methods

will be compared, as we are trying to find a suitable alternative that has similar performances, yet lends itself better for drawing inferences.

2) Methods in Detail

To give an impression of how the four methods perform and produce results, the descriptions of the methods are accompanied with demonstrations on an artificial dataset (i.e., toy data). This dataset was made with a pre-specified structure and the interest here was to see how well the methods approached this structure.

Besides the application of the four methods, a slightly adapted version of OTE with small trees (restricted to a small maximum node size) was applied. OTE with restricted trees was of experimental interest to try whether a reduced tree ensemble with shallow trees could be a possibly more interpretable alternative to OTE forests with full trees.

2.1) Specifications of data generation and model assessment measures

2.1.1) Data generation

The toy data was made with ten predictor variables $X_1, X_2, \ldots, X_{10}$, drawn from a standard multivariate normal distribution, and one outcome variable with two classes ($Y \in \{0, 1\}$). 1000 observations were drawn in total. Features X_9 and X_{10} were included as 'noise' variables that did not have any specific influence on distinguishing the two classes; only X_1 to X_8 were involved in producing the outcome. Of the latter variables in total four trees of size three were made, implying four two-way interactions. Each tree involved two features and as splitting variables either two positive or two negative split points. If the two-way interaction criterion was met, then the outcome 1 was drawn from a binomial distribution with probability $P(Y = 1) = .99$, which accounted for one of the terminal tree nodes. The other two leaves accounted for drawing outcome 0 with a probability of $P(Y = 0) = .99$ when either one or both of the thresholds were not exceeded. Combinations of thresholds were sought that yielded a proportion of $Y = 1$ between 1/3 and 1/2, while having as little overlap in tree rules producing $Y = 1$ outcomes as possible. The following rules $R_1, \ldots, R_4$ were specified:

$$R_1(X) = \begin{cases} Binom(1, .99) & \text{if } X_1 > 0.5 * X_2 > \Phi^{-1}(.625) \\ Binom(1, .01) & \text{else} \end{cases}$$

$$R_2(X) = \begin{cases} Binom(1, .99) & \text{if } X_3 \leq -0.5 * X_4 \leq \Phi^{-1}(.375) \\ Binom(1, .01) & \text{else} \end{cases}$$

$$R_3(X) = \begin{cases} Binom(1, .99) & \text{if } X_5 > 0.5 * X_6 > \Phi^{-1}(.625) \\ Binom(1, .01) & \text{else} \end{cases}$$

$$R_4(X) = \begin{cases} Binom(1, .99) & \text{if } X_7 \leq -0.5 * X_8 \leq \Phi^{-1}(.375) \\ Binom(1, .01) & \text{else} \end{cases}$$

$$Y = \begin{cases} 1 & \text{if } \sum_{j=1}^{4} R_j > 0 \\ 0 & \text{else} \end{cases}$$

where $\Phi^{-1}(.375)$ and $\Phi^{-1}(.625)$ denote percentiles of the standard normal distributions (with corresponding values -0.319 and 0.319 respectively). The outcome variable was computed by summing the Boolean outcomes of all rules and subsequently dichotomised by converting all values greater than 0 to 1. The trees $T_1, \ldots, T_4$ corresponding to the rules $R_1, \ldots, R_4$ are shown in Figure 1. After drawing data, the $y = 1$ proportions of the four trees produced were .127, .129, .134 and .125 respectively. The resulting $P(y = 1)$ was .42 (some of these $y = 1$ outcomes were made by overlapping rules from two (75) or three (9) trees).

The test set (of size 250) was generated with the same true model. It contained 153 instances of class $y = 0$ and 97 instances of class $y = 1$, hence $P(y = 1) = .39$.

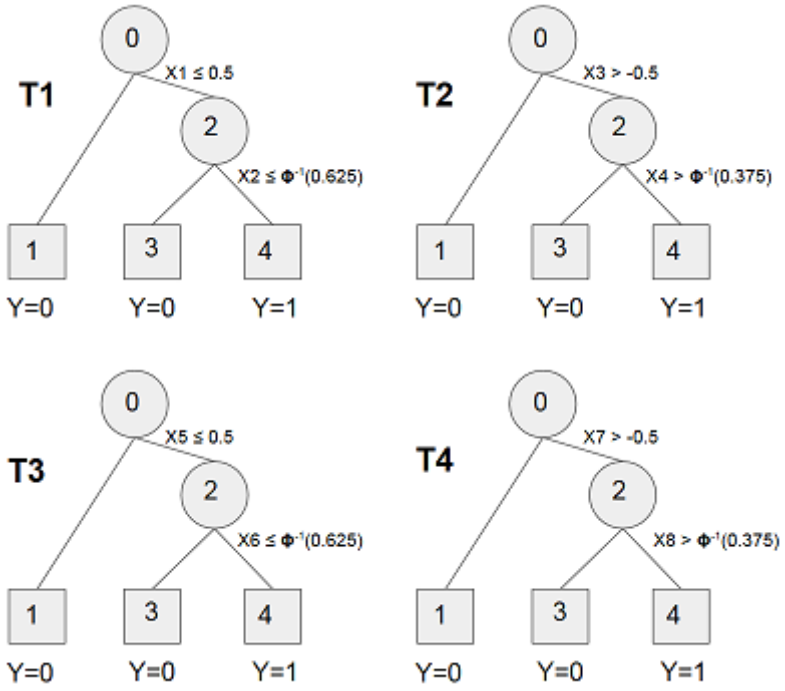


Figure 1: Trees with corresponding rules that determined the outcome of the toy data. If a split rule is true, then an observation goes to the left daughter node; else it goes to the right daughter node.

2.1.2) Measures of recovery performance

All methods were assessed on interpretability and their ability to recover the true effects present in the data. The measures used for this are divided into three categories: global recovery performance, specific recovery performance regarding correct variable interactions, and specific recovery performance regarding split points chosen. Global recovery performance was assessed by computing variable importances from trained models. The specific recovery performances for variable interaction and split points specifically were deduced from the first five ensemble members sorted on importance or weight. From these five most important ensemble members, the chosen splitting variables and corresponding thresholds (i.e., split points) were inspected and compared to the true effects specified in producing outcomes. Specific recovery performance could not be determined from random forest trees, as they are not sorted on importance values, nor on errors.

2.1.3) Measures of model performance

The performance of fitted models were assessed through predictions on the test set. Various measures were used, including accuracy, the Brier score, the AUC value and Press' Q; these are based on predicted classes or probabilities. Probabilities were rounded to the nearest integer (i.e., a cut-off of .5 was applied).

The most straightforward measure is the predictive accuracy. This is computed as the amount of correctly classified instances divided by the total sample size (as used in e.g., Maroco et al. 2011).

The Brier score (Brier 1950) is a measure of overall model performance, similar to R^2 used for continuous outcomes. Such measures depend on the squared differences between the predicted outcome $\hat{y}$ (which is the predicted class probability p for binary classifiers as used here) and the observed outcome y; the Brier score in particular is computed as $\hat{BS} = \sum_{i=1}^{N}(y_i - p_i)^2$ (Gerds, Cai, and Schumacher 2008; Khan et al. 2016). Its value ranges between 0 (perfect model) and 0.25 (a noninformative model).

The AUC statistic is an indication of the discriminative ability of a classification method. For binary classification this value equals the area under the receiver operating characteristic (ROC) curve. The ROC curve plots sensitivity (true positive rate) versus 1-specificity (false positive rate). The AUC value represents the probability that a randomly chosen subject with outcome $Y = 1$ (e.g., diseased) will be ranked higher than a subject with outcome $Y = 0$. The maximum possible value is 1 (Bradley 1997; Hastie, Tibshirani, and Friedman 2009).

The final performance measure for classifiers used here is Press' Q (Maroco et al. 2011). Press' Q is a statistic that determines if a classifier is able to classify better than chance alone. Press' Q is calculated as

$$Q = \frac{(N - nk)^2}{N(k-1)} \sim \chi^2(1),$$

where N is the total sample size, n the amount of observations correctly classified and k is the number of classes. Q is χ^2 distributed with one degree of freedom, under the null hypothesis that the classifier is not any

better than just chance. The critical value for Q is $Q_{crit} = \chi^2_{.05}(1) = 3.84$ at significance threshold $\alpha = .05$.

2.1.4) Software

Computations were performed in R versions 3.3.0 or 3.3.1 (R Core Team 2016), using the following R packages: randomForests (Liaw and Wiener 2002), OTE (Khan et al. 2015), nodeHarvest (Meinshausen 2015) and the RuleFit3 interface for R (Friedman and Popescu 2012). Optimization of the parameters where necessary was done with the caret package (Kuhn et al. 2016) and the ROC curves with their AUC values were calculated with the ROCR package (Sing et al. 2005). For all simulations random seeds were specified for reproducibility of the code experiments. All executed code is given in Appendix A.

2.2) Random Forests

2.2.1) Algorithm

On a training set with N observations, random forest grows CART trees on bootstrap samples of size N with the random subset constraint. This means that contrary to CART (or tree bagging), where by default all M variables are splitting variable candidates, S ($S \leq M$) variables are randomly chosen from all M variables and the best splitting variable has to be chosen from this subset. Tree growth stops when a maximum tree size or a minimum node size is reached. The default minimum node size for leaves is 1 in classification. The trees remain unpruned. A forest usually contains a few hundred trees; there is no guideline for the maximum amount of trees.

As mentioned before, random subset size S is the most important hyperparameter for random forests. For a dataset with M features, the default value in classification is assumed to be $S = \sqrt{M}$, although depending on the data this value is not necessarily optimal (Hastie, Tibshirani, and Friedman 2009). Also no paper was found that either explicitly confirms or argues the optimality of this value. Bernard, Heutte, and Adam (2009b) studied the influence of the hyperparameter S and found that the default settings of S can often be

sub-optimal, which is a reason to optimize S here by caret before fitting a model.

Random forest trees predict a class by aggregation: the class receiving the majority of votes by all trees together becomes the predicted class.

2.2.2) Model selection and model training

Minimum node size for the trees was set to 5 and was equal for all methods, except rule ensembles, as the default value was not applicable in node harvest. In node harvest the default value is 10 and Meinshausen (2010) recommends a value of at least 5 to achieve good results.

The optimal value for hyperparameter S was selected from the following candidates: 1, $\lfloor\sqrt{10}\rfloor = 3$ (i.e., the default; $\lfloor^2 log(M+1)\rfloor =^2 log(11)$, as used in Breiman (2001), gives the same value), $M/2 = 10/2 = 5$, 8 (the number of specified interactions) and $M = 10$. The optimal S found by caret using repeated bootstrapping (repeated 25 times) on the training set was $S = 5$ (accuracy rate: .929). Thus the eventual random forest model was trained with a minimum node size of 5 and random subset size of 5 (Table 5).

2.2.3) Global recovery performance

The variable importance values computed for random forests are based on the Gini index. The Gini index is defined as:

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}),$$

with $\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_M} I(y_i = k)$ being the proportion of observations in node m belonging to class k. Variable importance values are calculated from the total improvement in node impurities, measured by the Gini index, gained by including a certain splitting variable in every tree. These values are then averaged over all trees to yield the importance measures (Hastie, Tibshirani, and Friedman 2009), i.e., the mean decrease in

Table 1: Variable importances per variable for all methods. These are measured in mean decrease of Gini index $\overline{\Delta(G)}$ for random forests and OTE and are rounded to one digit. For node harvest variable importances are based on node weights, for rule ensembles variable importances are based on relative importance measures; both are expressed as proportions $\hat{p}$ (and hence rounded to two digits).

Variable	RF	OTE1	OTE2	NH	RE
x_1	53.7	49.8	11.2	.43	1.00
x_2	54.6	51.7	1.1	.33	.71
x_3	50.0	51.3	8.8	1.00	.83
x_4	49.8	48.1	13.4	.87	.89
x_5	59.9	55.0	25.3	.51	.64
x_6	56.3	54.4	0	.53	.74
x_7	56.3	57.2	3.2	.45	.84
x_8	55.2	49.6	4.0	.44	.88
x_9	13.0	11.7	0	0	0
x_{10}	11.9	9.1	0	0	0

Gini index is reported ($\overline{\Delta(G)}$). A larger value $\overline{\Delta(G)}$ indicates that a variable is more important.

Although trees in a random forests are said to capture interactions, it is not possible to compute interaction measure estimates similar to individual variable importance measures. Wright, Ziegler, and König (2016) have looked into this and attempted to produce interaction importance estimates, but failed: they found that interaction effect estimates are masked by marginal effects. Moreover, it is quite cumbersome to differentiate marginal effects from interaction effects.

The estimated variable importances for the fitted random forest model are summarized in Table 1 and plotted in Figure 2a. They show that variables x_1 to x_8 were regarded as most important and x_9 and x_{10} as least important, as there were large differences in mean decrease in Gini index between x_9 and x_{10} with the other variables. However, the values for x_9 and x_{10} were not (close to) zero, indicating that they have sometimes been selected as splitting variables; random forests do indeed not ignore any variable (Hastie, Tibshirani, and Friedman 2009). x_1 to x_8 had quite similar variable importance values: the model seemed to regard them of being of similar importance in predicting outcomes.

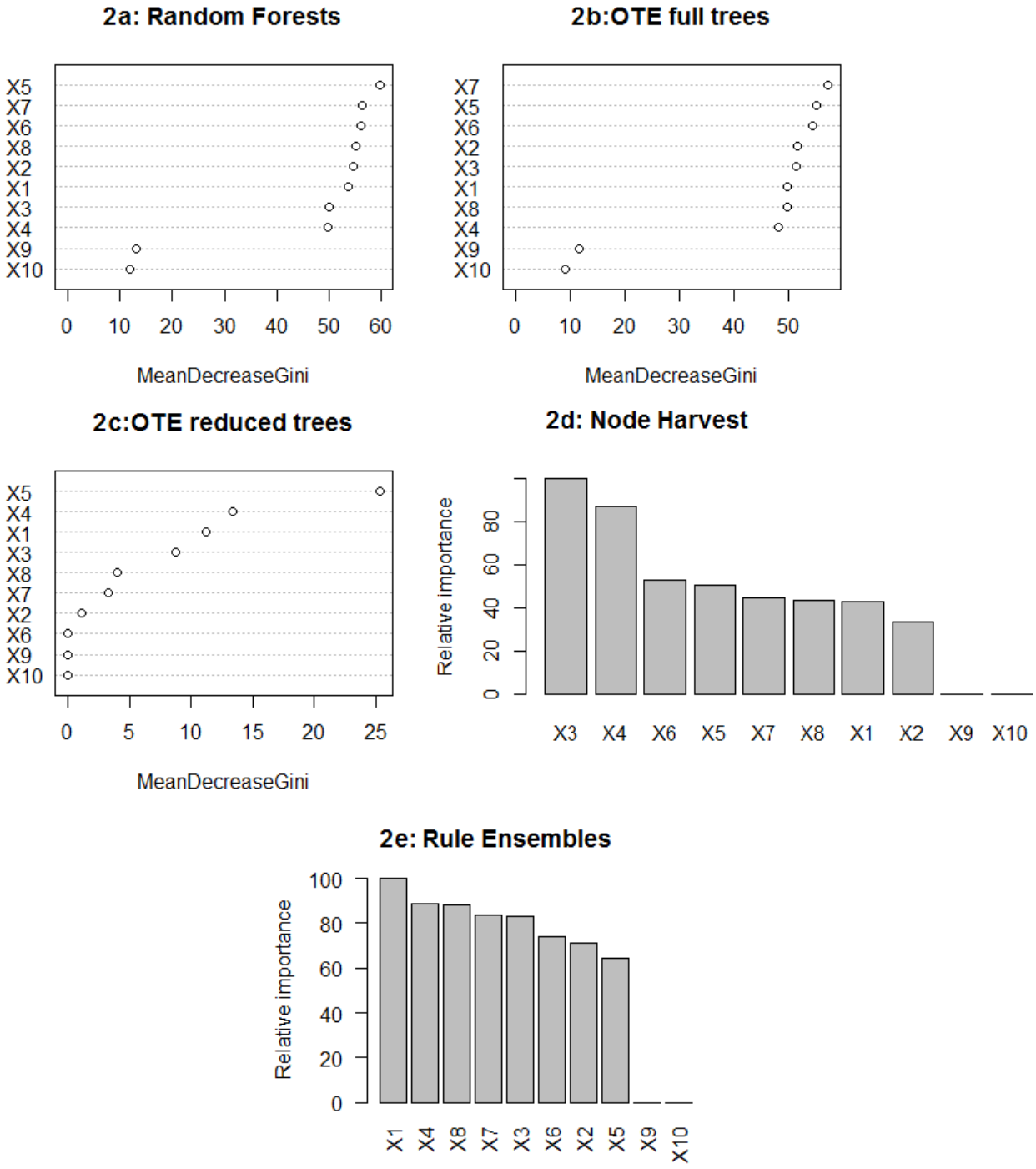


Figure 2: Variable importance plots calculated from the fitted models of random forests (a), OTE with full trees (b), OTE with restricted trees (c) (a-c using the Gini index), node harvest (d) and rule ensembles (e) (d-e with relative importance measures expressed as percentages).

2.3) Optimal Trees Ensemble

Optimal trees ensembles is a recently proposed promising method that selects the first F strongest and most diverse trees of a random forest to reduce the total forest size. This is beneficial for saving memory space and speeding up predictions. The resulting ensemble is less complex, yet is able to capture meaningful structures in the data like more complex methods are. In Khan et al. (2016) OTE performed comparable to or slightly better than random forests, which proves that it is not necessary to use all information contained in a full tree ensemble to do accurate predictions.

2.3.1) Algorithm

Optimal trees ensembles starts off in a similar way as random forests. After the training data is split randomly into a growing set L_G and a separate validation set L_V, the random forest is grown on L_G. The validation set L_V is used to help construct the ensemble of optimal trees. The first F important trees from the forest, which is a certain proportion p of all trees, are selected based on their strength and diversity. In classification, tree strength is assessed by classification errors on the out-of-bag (OOB) observations from L_G. Trees are then ordered on ascending OOB error values. Then the diversity check is done by adding candidates one-by-one from the first F best performing trees to the optimal ensemble. Subsequently added trees are kept in the ensemble if their addition improves the Brier score (i.e., it decreases) from predictions with L_V compared to the current ensemble composition. Predicted classes are determined by majority votes, similar as random forests (Khan et al., 2016).

2.3.2) Model selection and model training

For model specification, certain options applied in random forests previously were adopted here as well: the optimal S was 5, the minimum node size was 5 and the initial ensemble size was 500 (Table 5). By default 20% of the trees of the initial ensemble are selected as candidates for the optimal ensemble, so 100 trees are sorted and added one by one to construct the optimal ensemble. The final ensemble size would then

be 100 trees or less. However, it is possible to adjust this percentage to make possibly smaller or bigger final ensembles. Different values of percentages were tried: 20%, 15%, 10%, 5% and 1%; 100 train and test datasets (with the same toy data structure) were generated and OTE models with every percentage value were fitted to every training set. Performance was assessed on the test sets. Four paired t-tests showed that down to 10% there was no significant difference in model performance compared to 20% (Bonferroni-corrected $p_{adj} = 1.000$). Lower percentage values had significantly lower accuracy values (20% vs 5% $p_{adj} = .016$ and 20% vs 1% $p_{adj} = .000$). Hence a value of 10% was selected to fit a model on the training set, such that the final optimal ensemble was chosen from 50 trees. The final OTE consisted of 27 trees (Table 5).

2.3.3) OTE model 1 with full trees

2.3.3.1) Recovery performance and interpretation

2.3.3.1.1) Global recovery performance

Global recovery performance measures for OTE were computed in the same way as for random forests (i.e., based on the Gini index), because the OTE method is directly based on random forests. The variable importances of x_9 and x_{10} were least important, based on the final OTE (Table 1, Figure 2b).

2.3.3.1.2) Specific recovery performance

As a final ensemble of optimal trees consists of trees sorted on smallest OOB error (and improvement of the Brier score), the first few trees in the ensemble are assumed to be the strongest trees and the best discriminators. Trees $t_1, t_2, \ldots, t_5$ were selected from the fitted OTE, from which the splitting variables and split points were inspected. As the true patterns in the toy data only specified rules for predicting class 1, only rules ending in a terminal node predicting class 1 are given (see Appendix B). Some correct interactions were captured in these rules, but many incorrect interactions and thresholds were included as well. Furthermore, every tree had a high amount of rules and most of these rules consisted of interactions of very high orders.

This does not improve interpretation much compared to random forests.

2.3.4) OTE model 2 with restricted tree size U

A maximum tree size $U = 3$ was chosen, as the true structure underlying the data was based on trees with that amount of leaves.

2.3.4.1) Recovery performance and interpretation

2.3.4.1.1) Global recovery performance

Overall, variable importance value differed a lot with variable importance measures for the random forests and full tree OTE model (Table 1; Figure 2c). x_6 even got a 0 value, just as x_9 and x_{10}.

2.3.4.1.2) Specific recovery performance

The first five strongest trees $t_1, t_2, \ldots, t_5$ included in the final ensemble are displayed in Figure 3. All trees except the fourth found correct interactions with split point values close to the true threshold values. Unfortunately, not all interactions specified were included within these five strongest trees. The interaction between x_3 and x_4 occurred three times. The fourth tree specified an interaction between x_5 and x_2. Peculiarly, the two-way interaction led to two class 0 predictions and only a main effect involving x_5 led to a leaf predicting class 1.

As selecting shallow trees to form an optimal ensemble of trees did not lead to better information recovery performance or predictive performance (see Section 2.6), we did not further investigate this adaptation of OTE in this study.

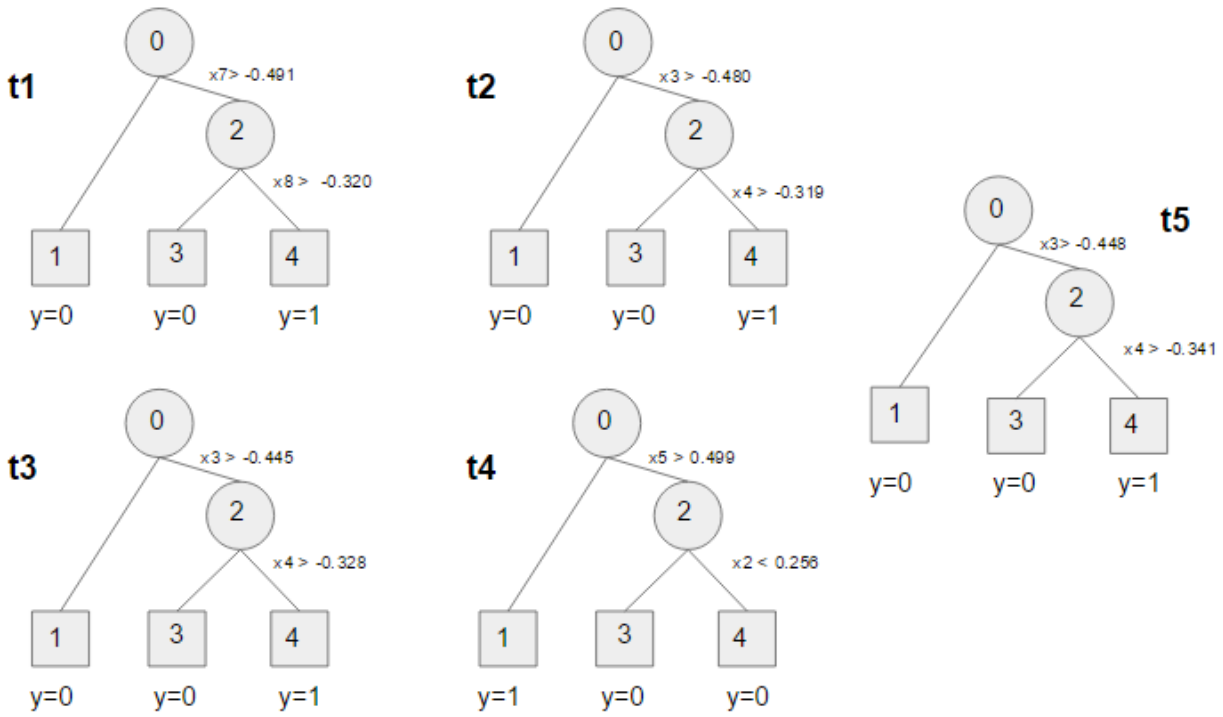


Figure 3: Five strongest trees $t_1, t_2, \ldots t_5$ selected from the optimal trees ensemble with restricted tree size. The rules of the internal nodes are noted; if true, then the left branch is followed.

2.4) Node Harvest

Node harvest (Meinshausen 2010) is a method in which a large set of nodes, or splitting rules, are generated from random forest trees. With a quadratic programming problem the most important nodes, i.e., those giving the lowest prediction error on the training sample, are selected and given weights. A new observation belonging to a few of the nodes in the final ensemble gets a weighted prediction from the involved nodes. From this method not only decision tree-style prediction rules are available but also outputs showing the nodes with corresponding variables and their thresholds. Other characteristics of node harvest models are that they have sparse solutions, they are able to handle missing data and capture interactions.

2.4.1) Algorithm

Node harvest makes an ensemble of nodes instead of trees, though nodes are drawn from random forest trees. Instead of growing trees on bootstrap samples of size N, trees are grown on subsets of data of size

$N/10$, reducing computation time. From the trees nodes are randomly selected that correspond to a specified maximum interaction order or lower and comply to a minimum node size. Nodes with identical rules are removed. These steps are repeated until a maximum initial node ensemble size is reached; this is the initial node ensemble. Then the most important nodes are sought with the node harvest estimator to form the final ensemble. The node harvest estimator is a convex optimization problem that finds the optimal vector of weights to select a small subset of nodes for the final node ensemble. This convex loss function for binary classification is a least-squares loss. The weights $\mathbf{w}$ have values between 0 and 1. In the final node ensemble a root node, which contains all observations, is always included and usually gets a small weight. Node harvest predicts differently than random forests and optimal trees ensembles but more similar to the regression prediction rule of CART. Predicted values of node harvest are weighted averages, which are calculated from the weights of the involved nodes and the average responses in them. This accounts for both regression and classification models. Average node responses in the case of binary classification represent the proportion of observations belonging to class $y = 1$. Hence, predicted values in classification correspond to predicted probabilities and can have the values $\hat{y} \in [0, 1]$ (Meinshausen 2010). Similar to the recursive partitioning sequence of a (CART) tree, with node harvest it is possible to search the corresponding nodes into which one observation falls.

2.4.2) Model selection and model training

Specifications for the node harvest model were a minimum node size of 5 and maximum interaction order of 2 (Table 5). Node harvest is also included in the caret package for optimizing maximum interaction depth. However in previous simulations (not shown here) the results were unsatisfactory, as often the interaction orders found were more complex than specified. Hence the default interaction depth of two was applied for node harvest. This interaction order is shown to perform generally well and usually it is unnecessary to allow for more complicated interactions (Meinshausen 2010). Nevertheless, the main motivation for choosing so in this particular application was that the tree structures underlying the data were of depth 2 and the interest was whether node harvest managed to extract the rules of these tree structures. The initial node ensemble

size was set to 1500. The two-way interaction trees underlying the outcomes of the toy data were of size 3 and previously random forests were grown up to 500 trees, hence 1500 nodes should be yielded in total to get an ensemble equivalent to a random forest ensemble of size 500. The predicted class probabilities were rounded to the nearest integer to get the predicted classes to compute accuracy rates.

2.4.3) Recovery performance and interpretation

2.4.3.1) Global recovery performance

Variable importance measures are not described in the original node harvest paper (Meinshausen 2010), nor is there a function available in the corresponding R package to compute such a measure. However, to allow for a comparison of methods that was as complete as possible, variable importances were computed for the node harvest model based on the variable importance formula used in rule ensembles (as described in Section 2.5.3.1, formula (2)). The formula was adapted to:

$$J(x_l) = \sum_{x_l \in q_k} \frac{w_k(x)}{m_k}, \tag{1}$$

where the node weights $w_k(x)$ substitute the term for rule importances $I_k(x)$ (Friedman and Popescu 2008). The term m_k is the amount of variables contained in a node q_k, analogous to the amount of variables in a rule r_k. All importance values $J(x_k)$ are scaled such that the largest value gets a relative variable importance of 1.

Variable x_3 had the highest importance (Table 1; Figure 2d), indicating that this variable was either chosen most often or occurred frequently in nodes with higher weights. The five nodes with highest weights were inspected in more detail (Table 2) and they indeed confirmed that four of the five most important nodes contained x_3. x_4 was the second most important variable. The variable importance values of x_9 and x_{10} were 0, these variables were apparently not included in members of the final ensemble.

2.4.3.2) Specific recovery performance

Table 2: Five nodes q with the highest weights selected from the final node harvest ensemble. Per node the variable interactions and corresponding split points, node weight $\mathbf{w}$, size of training set observations contained in a node (n_j; also represented as support $s_j = n_j/N$) and predicted average value ($\hat{y}_j$) are given.

Node q_j	Rule	$\mathbf{w}_j$	n_j	s_j	$\hat{y}_j$
q_{49}	$x_3 >$ -0.480 & $x_5 \leq 0.467$	0.177	477	.48	.247
q_{22}	$x_3 >$ -0.480 & $x_5 > 0.467$	0.177	210	.21	.595
q_5	$x_3 \leq$ -0.480 & $x_4 \leq$ -0.335	0.177	119	.18	1
q_{15}	$x_3 \leq$ -0.480 & $x_4 >$ -0.335	0.177	194	.18	.309
q_9	$x_5 > 0.498$ & $x_6 > 0.318$	0.156	128	.13	.992

The first five important nodes (Table 2) were inspected for more detailed interpretation, these nodes were sorted on descending weights. Only two of these nodes were entirely correct regarding prediction rules and predicted values: nodes 5 and 9 had average responses that were (close to) 1. They captured the correct interactions x_3&x_4 and x_5&x_6 respectively and approximately contained the correct amount of observations (n_j or support proportion s_j) accounting for these interactions (i.e., the tree proportions mentioned in subsection 2.1.1). The corresponding threshold values were also very close to the true split points. Nodes q_{49} and q_{22} had incorrect two-way interactions between x_3 and x_5, although their split points were very similar to those specified in the toy data. Node q_{15} was similar to node q_5 regarding the interaction and split points, although the direction of the split points was different, yielding a different average class proportion. This could be compared to the leaves of a decision tree, where nodes q_5 and q_{15} could be the daughter nodes of one common parent node that specifies a two-way interaction between x_3 and x_4. For this decision tree, node q_5 would only have observations belonging to class 1 and node q_{15} would contain about 70% of class 0 and only 30% of class 1, i.e., one node would predict class 1 and the other 0.

A plot of all nodes of the final ensemble can be requested as well (Figure 4). This gives an impression of the average outcomes predicted by the nodes with corresponding prediction rules. The values on the x axis correspond to the average prediction of that node, the y axis the amount of observations contained in one node. Circle sizes correspond to weight sizes: larger nodes have larger weights. Still, this plot is quite chaotic, which makes this plot not a very convincing visualization tool to aid interpretation.

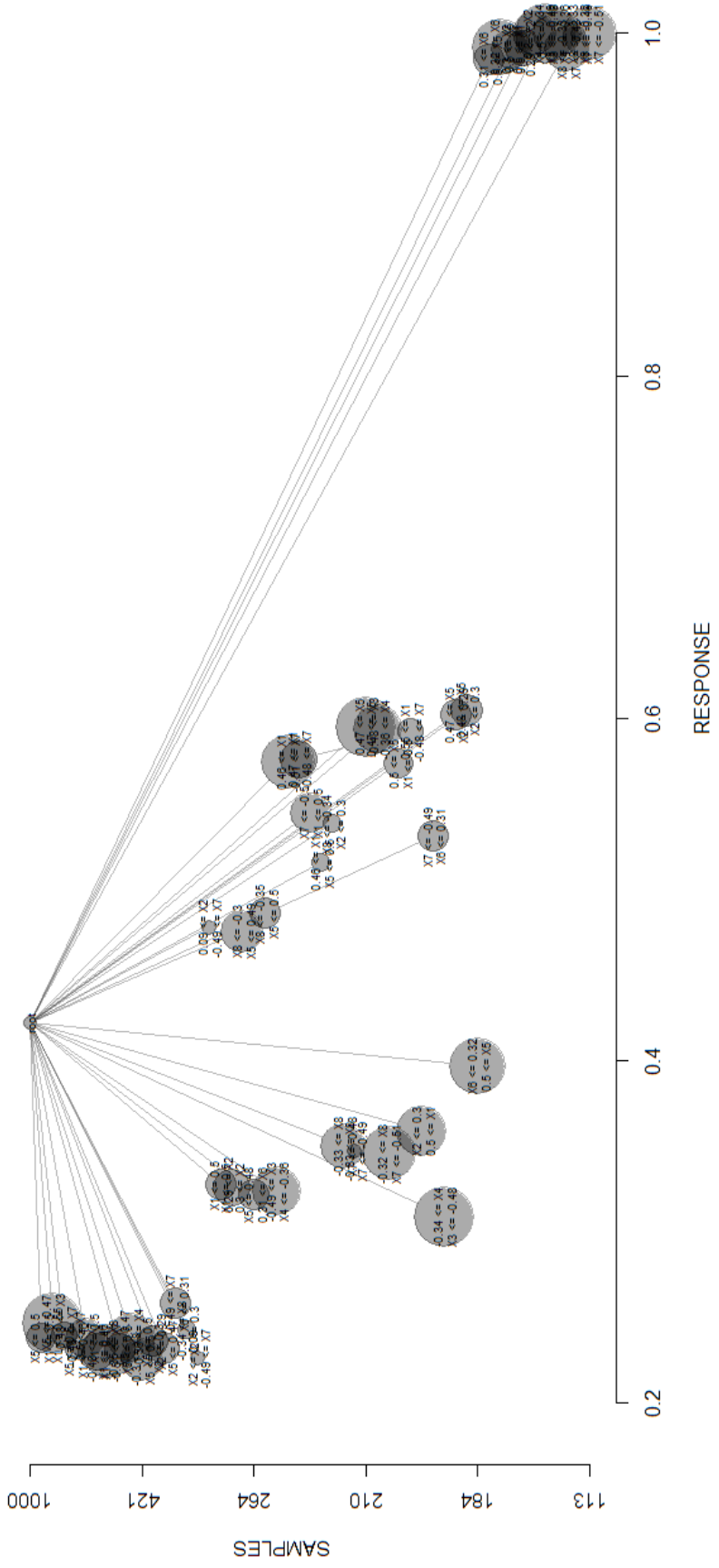


Figure 4: Plot of the nodes in the node harvest model fitted to the toy data. These nodes received nonzero weights. Their size is relative to their weights. The x-axis represents the average response values of the nodes, i.e., the average outcome of the training observations belonging to that node. The y-axis shows how many training observations are included in a particular node (i.e., n_j).

2.5) Rule Ensembles

Rule ensembles (Friedman and Popescu 2008) is another interpretation-focused ensemble method. A tree ensemble, with slight dependency between consecutive trees, is grown and every individual tree is seen as a set of rules, with every node in a tree corresponding to a rule. The optimal linear combination of these nodes/rules is sought by solving a Lasso-like equation for a set of parameters that specify particular linear combinations in an ensemble. A prediction is a linear combination of a set of rules, which are specified with binary indicator variables representing whether a rule applies an observation or not. The prediction rules, that can be requested, may give insight in variables, thresholds, and interactions between variables.

2.5.1) Algorithm

Rule ensembles uses trees and/or linear models as base learners to construct an ensemble. There will be no elaboration on the utilization of linear base learners, as this study focusses on the role of trees in an ensemble. Rule ensembles uses class labels $Y \in \{-1, 1\}$ in binary classification problems. Tree ensemble generation is based on the importance sampled learning ensemble (ISLE) method, described in Friedman and Popescu (2003). Base learners are grown on randomly drawn subsamples of size $\iota = N/2$ and with a slight dependency between subsequently grown trees. Dependency is determined by shrinkage parameter $\nu = 0.01$ ($\nu = 1$ represents full dependency on previous base learners such as in Ada.Boost, and $\nu = 0$ no dependency such as in random forests). Tree sizes U are drawn from an exponential distribution with a user-specified mean, representing an average generated tree size $\bar{U}$, which is equivalent to the maximum interaction order of node harvest. The default value is $\bar{U} = 8$. All nodes of a tree produce a rule; tree generation stops when a certain maximum amount of rules is yielded. The rule ensemble takes the general ensemble model form:

$$F(x) = a_0 + \sum_{m=1}^{M} a_m f_m(x),$$

where M is the ensemble size, $f_m(x)$ is an ensemble member and $F(x)$ is an ensemble prediction made by a

linear combination of every prediction created by individual ensemble members. A loss function with lasso constraint is solved to find the coefficients a_m for the ensemble members $f_m(x)$. The optimal lasso parameter λ is found through internal three-fold cross-validation. The constraint for finding the coefficients uses a squared error ramp loss (Friedman and Popescu 2003):

$$L(y, F) = [y - min(-1, max(1, F))]^2.$$

Eventually many rules (~80% to 90%) have coefficients set to 0, causing them to be removed from the rule ensemble and thus strongly reducing the total ensemble size. The coefficients represent the direction of the effect of the rule: in classification, the sign of the coefficient corresponds to the predicted class (i.e., 1 or -1). The size of the coefficient reflects the relative importance of the rule as well as the support of the rule from training data observations (i.e., how many training observations fall into this rule). The rules for a new observation are determined by indicators: if an observation belongs to a rule, a 1 is scored, otherwise a 0. The predicted score is then a summation of the coefficients of those rules. The resulting value is a log-odds, which is an indication of confidence in class prediction: the larger its absolute value, the more certainty there is about the predicted class. The predicted class is the sign of this value (Friedman & Popescu 2008).

2.5.2) Model selection and model training

The average tree size value was set to $\bar{U} = 3$ (Table 5), which corresponds to one two-way interaction such as in the four trees determining the outcomes of the toy data (Section 2.1.1). The initial ensemble size was set to 1500, which is equivalent to 500 trees with three leaves. Subset sample size ι and dependency parameter ν retained their default values (i.e., $\iota = N/2$; $\nu = .01$) as recommended in Friedman and Popescu (2008). In rule ensembles it was unfortunately not possible to specify minimum node/rule size.

2.5.3) Recovery performance and interpretation

2.5.3.1) Global recovery performance

Variable importances for a rule ensemble are not based on a purity measure such as in random forests. Rather, they are based on the importances of individual rules (see equation 3) to which a variable belongs and how many variables a rule contains in total:

$$J(x_l) = \sum_{x_l \in r_k} \frac{I_k(x)}{m_k}. \tag{2}$$

For a variable x_l, all importances of individual rules I_k containing that variable are summed and divided by the total number of variables m_k contained in a rule (formula 35 in Friedman & Popescu 2008; formula modified to include only rule base learners). Thus if a variable appears more often, it is found to be more important.

Table 1 and Figure 2e show the relative variable importances for variables x_1 to x_{10}. These values were 0 for both x_9 and x_{10}, hence these were not included in any rules of the final rule ensemble. All other variables had quite high values of relative importance. x_1 was the most important, i.e., it occurred most often in the most important rules (Friedman and Popescu 2008).

Not only was it possible to extract variable importances for the variables, interaction strengths between variables could be extracted as well (Table 3). Although x_1 to x_8 had some interactions with each other (except with x_9 and x_{10}), it is evident that the strongest two-way interactions were between variables paired in the toy dataset. x_9 and x_{10} had no interactions with any of the variables, confirming that they never occurred in any two-way interaction rules. However, they had full interactions (a value of 1) with each other according to the output of the program.

The rule ensembles interface can also produce three-way interaction values; these could also be requested here as with an average tree size of $\bar{U} = 3$ often some larger trees are grown. However they were of less interest in

Table 3: Relative interaction strength between variables. The bold values emphasize the highest interaction strength for a variable.

Variable	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
x_1	-	**.65**	.10	.16	.02	.02	.09	.01	0	0
x_2	**.65**	-	.02	0	.02	0	.04	.07	0	0
x_3	.10	.02	-	**.69**	.05	0	.02	.05	0	0
x_4	.16	0	**.69**	-	.01	.05	0	.04	0	0
x_5	.02	.02	.05	.01	-	**.70**	.02	.02	0	0
x_6	.02	0	0	.05	**.70**	-	.04	0	0	0
x_7	.09	.04	.02	0	.02	.04	-	**.62**	0	0
x_8	.01	.07	.05	.04	.02	0	**.62**	-	0	0
x_9	0	0	0	0	0	0	0	0	-	1
x_{10}	0	0	0	0	0	0	0	0	1	-

the current situation and the values were expected to be small since mainly two-way interaction rules are found, hence these values are left out here.

2.5.3.2) Specific recovery performance

The selected rules from the final ensemble can be sorted by rule importance. Rule importance I for a rule j is calculated from the coefficient value $\hat{a}_j$ and support s_j:

$$I_j = |\hat{a}_j|\sqrt{s_j(1-s_j)} \tag{3}$$

(Friedman and Popescu 2008). The support s of a rule is the proportion of observations of the training set that apply to that rule.

The first five rules with highest importances of the fitted model are given in Table 4, together with their coefficients and support values. Rules 1 to 5 captured all two-way interactions specified in the toy dataset that predicted $y = 1$ (the signs of their coefficients were positive). The corresponding threshold values for all the variables were very similar to the true split points. Moreover, the support values of the rules were almost the same as the proportion of $y = 1$ outcomes within the four trees separately (these values were between .125 and .135, as described before Section 2.1). Rules 3 and 5 were similar, indicating that the model probably selected some interactions more than once. However, their split points slightly deviated and the

Table 4: Five rules of the fitted rule ensemble model with highest relative importances I. Per rule the variables or interactions and threshold values are given, as well as the support s and coefficient values a.

Rule order	Rule	s	a	I
1	$x_5 > 0.498$ & $x_6 > 0.318$	.13	4.43	1.00
2	$x_1 > 0.502$ & $x_2 > 0.308$	.12	4.11	.90
3	$x_3 <$ -0.486 & $x_4 <$ -0.331	.12	3.53	.77
4	$x_7 <$ -0.491 & $x_8 <$ -0.320	.12	2.69	.58
5	$x_3 <$ -0.439 & $x_4 <$ -0.307	.13	1.96	.44

support was not entirely the same. Rule 3 was more important according to the model than rule 5, despite that rule 5 had more support. Rule 5 had threshold values that deviated more from the true split points in combination with more support, probably because the true split points of the x_3&x_4 interaction could not explain all outcomes of y which were made by the combination of four trees.

In short, the final rule ensemble with its individual members were very accurate in finding the true underlying model regarding interactions, split points and true proportion of data supporting these true rules.

2.6) Model performances

Table 5 summarizes the specifications for the fitted models, their initial and final ensemble sizes and obtained predictive performances as assessed on the test set. Figure 5 shows the corresponding ROC curves for all five fitted models to the toy data.

Except for the OTE model with reduced trees, every model performed well. Random forests, OTE with full trees and rule ensembles were comparable in model performances. Node harvest performed quite well, though not as good as the first three mentioned methods. The Brier scores indicated that for all methods there could be some improvement regarding predicted probabilities. It implied that the the predicted probabilities were not always close to the true class values or that some cases were misclassified. For node harvest this can be deduced from the node plot in Figure 4, where there were no nodes predicting an average proportion of 0, but only proportions $\geq .2$.

Table 5: Summary of fitted models (parameters, initial and final ensemble sizes) and their performances on the test set, calculated with four different measures.

Methods with parameters specified	Initial ensemble size	Final ensemble size	Accuracy rate	Brier score	AUC value	Press' Q
Random Forests ($S = 5$, node size=5)	500	500	.960	0.061	0.960	211.60
OTE ($S = 5$, prop=0.1, node size=5)	500	27	.952	0.058	0.957	204.30
OTE2 ($S = 5$, prop=0.1, node size=5, $U = 3$)	500	19	.696	0.189	0.794	38.42
Rule Ensembles ($\bar{U} = 3$, maximum number of rules=1500, method=rules)	1500	59	.964	0.038	0.946	215.30
Node Harvest (maximum interaction order=2, nodes≈1500, node size=5)	1618	51	.908	0.152	0.939	166.46

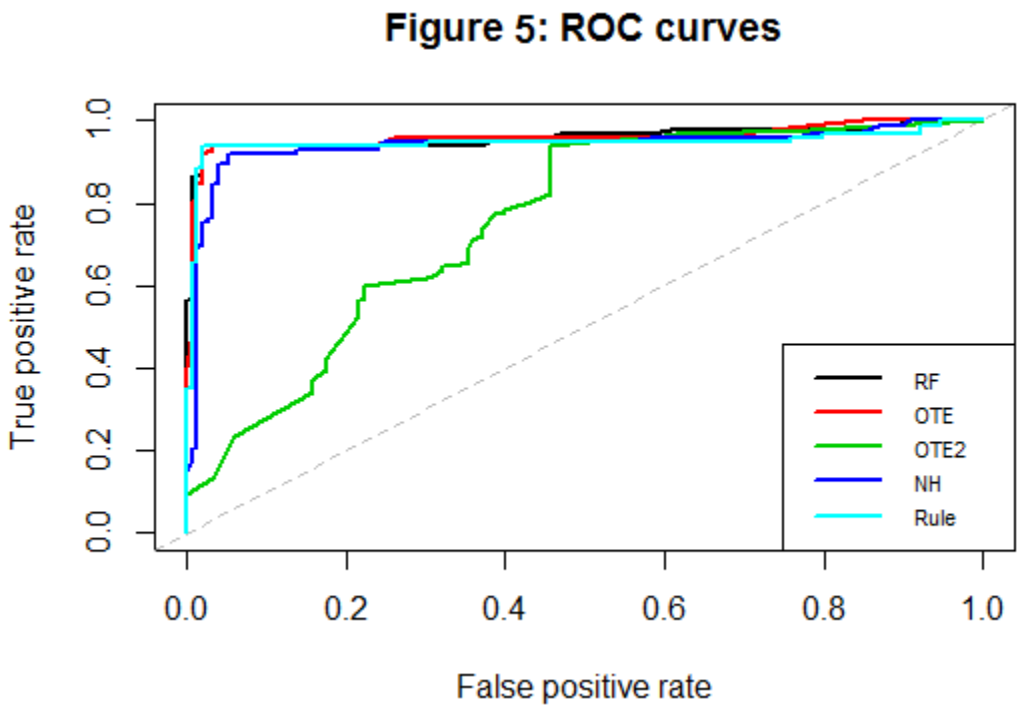


Figure 5: ROC curves for the models fitted with random forests (RF), optimal trees ensemble with full trees (OTE), optimal trees ensemble with restricted trees (OTE2), node harvest (NH) and rule ensembles (Rule). The dashed grey line represents what the ROC curve would be for uninformative models.

OTE with reduced trees performed worst (also confirmed by Fig. 5, where the corresponding ROC curve is furthest away from the upper left corner), although still better than chance (Brier score=.189, Press' $Q = 38.42$). About one third of the outcomes were misclassified (accuracy rate of .696). The confusion matrix of the fitted model showed that of the 153 class 0 outcomes, 35 observations got misclassified as 1 (22.88% false positives). Conversely, 41 of the 97 outcomes of class 1 were wrongly predicted as 0 (42.27% false negatives). This indicated that class 1 instances were more often incorrectly predicted than class 0 instances, despite that the true outcomes were mainly based on clear rules predicting class 1.

The relative reduction in ensemble sizes were 5.40% and 3.80% for OTE with full and reduced trees respectively, 3.93% for rule ensembles and 3.40% for node harvest. For OTE with full trees and rule ensembles the vast reductions in ensemble size did not seem to be at the expense of model performance compared to random forests' ensemble size.

2.7) Global summary

The best classifiers were random forests, OTE and rule ensembles. The latter two methods achieved this with strongly reduced ensemble sizes, proving that accurate ensembles can consist of less members.

As already stated in the introduction, random forests is an excellent predictor, but regarded as a black box predictor. For a single observation, it is difficult to trace back how its predicted class is determined. Inspecting individual random forest trees is possible. However, this does not give a fair impression of how outcomes depend on variables and how variables interact, as one tree can have hundreds of nodes and subsequent splits are made with certain randomness. Furthermore, there are no additional plots or outputs available for random forests, so interpretation remains limited to the variable importance plots where only main effect importance can be assessed.

OTE produces smaller final tree ensembles than random forests, consisting of the strongest trees. The global recovery performance of the full OTE model was good; the uninvolved predictors got the smallest variable importance values. However, decomposing the strongest trees of an OTE did not help enhancing interpretation with unpruned trees, as prediction rules were highly complex. This does not give OTE (with full trees) any edge in interpretation compared to random forests.

OTE could not be improved any further for interpretation by constructing an OTE with shallow trees. OTE with reduced tree depth had the worst global recovery performance. If the structure underlying the data would not be known beforehand, some variables could unjustly be interpreted as being irrelevant for prediction, as they have a chance to be excluded from the final model. While a method like rule ensembles does excel with very simple ensemble members, reducing tree complexity did not yield any improvements for OTE. A reason for this is that such trees contained one almost entirely pure leaf (i.e., the leaf satisfying the rule) and the two remaining rules were less pure, as they contained the (mixed) remaining class outcomes produced with other rules.

Node harvest was a reasonably good predictor, but was less accurate than random forests. Making nodes

based on average outcomes (i.e., class proportions) in a binary classification setting, rather than using a purity criterion, could have harmed predictive accuracy and discriminative ability or prediction certainty. Computing variable importance values showed that the model correctly identified the most important variables involved in distinguishing classes. The node harvest model was also able to find the split points of the variables quite well. However, the variable interactions were not always correct.

Rule ensembles proved that a strongly reduced ensemble with very compact ensemble members can contain not only the right information, but also enough information needed to construct a highly accurate ensemble. It had a good global recovery performance and the most informative specific recovery performance outputs. In this demonstration, rule ensembles was the clear overall winner: it was among the most accurate models (which included the benchmark random forests) and it retrieved almost the exact underlying structure of the data.

3) Simulation

In this section the information recovery performance and predictive performance of all methods in various settings were assessed in a larger simulation study. There were three design factors: training set size, proportion of class outcome labels, and error. The interest was how these factors influenced recovery and predictive performance.

3.1) Simulation set-up

3.1.1) Design factors

The first design factor, training set size n_{train}, had values 250, 500, 1000 or 5000. Noise *error* had values .5 and .0 and was derived from Nagelkerke's R^2 values (Nagelkerke (1991); explained in Section 3.1.5). Nagelkerke's R^2 was computed from logistic regression models fitted to the simulated data. An essentially noise-free model - a model perfectly explaining the observed variation (Nagelkerke's $R^2 = 1$) - has a noise value value of $error = 1.0 - R^2 = .0$. This value was achieved by fitting a model to data that had been generated in an (almost completely) deterministic way. Hence, for these cells data was generated by a rule or tree structure as in Section 2. A value of .5 means that only half of the variation is explained (Nagelkerke's $R^2 = .5$), implying that more noise is involved. Data for these cells were made with the underlying logistic regression model with varying weights (explained in subsection 3.1.2).

The third factor, $P(Y = 1)$, was fixed at .5 or .1. .5, the balanced class level, allowed for an equal amount of information available for finding effects defining either one of the classes. .1, the unbalanced class level, could represent proportions of for example diseased and healthy people, where disease prevalence is low (10%) in the population.

The full factorial design resulted in $4 * 4 * 2 = 16$ cells in total. 100 training datasets were generated in each cell of the design. To each dataset all the different methods were applied (see 3.1.3). In addition, a test

dataset of $n_{test} = 250$ was generated for every training dataset with similar structures.

3.1.2) True underlying model for data generation

Data for ten X variables were drawn from a multivariate normal distribution with $\boldsymbol{\mu} = 0$. Contrary to the set-up in Section 2, the interactions of X_3 with X_4 and X_5 and X_6 were substituted by interactions X_9 with X_4 and X_5 with X_{10} respectively. This was a minor precaution to ensure that the methods did not prefer variables or interactions in the order of which they appeared in the dataset. Interacting X variables were drawn with some dependency on each other: the covariance of a pair of interacting variables was set at 0.5. The diagonal values of the covariance matrix (i.e., the variances) remained 1.

The outcomes Y were made using the tree-based model ($error = .0$) or a logistic regression model ($error = .5$). The tree-based model used for outcome generation was of identical form as described in Section 2, except with different thresholds. The general forms of the tree rules R_j are:

$$R_1(X) = \begin{cases} Binom(1, .99) & \text{if } X_1 > a * X_2 > a \\ Binom(1, .01) & \text{else} \end{cases}$$

$$R_2(X) = \begin{cases} Binom(1, .99) & \text{if } X_9 \leq b * X_4 \leq b \\ Binom(1, .01) & \text{else} \end{cases}$$

$$R_3(X) = \begin{cases} Binom(1, .99) & \text{if } X_5 > a * X_{10} > a \\ Binom(1, .01) & \text{else} \end{cases}$$

$$R_4(X) = \begin{cases} Binom(1, .99) & \text{if } X_7 \leq b * X_8 \leq b \\ Binom(1, .01) & \text{else} \end{cases}$$

$$Y = I(\sum_{j=1}^{4} R_j > 0).$$

Thresholds a and b were fixed beforehand and depended on whether the class proportion levels were balanced or not. Of the design cells for which data was drawn from the tree-based models (i.e., $error = .0$), four cells had thresholds $a = 1.35$ and $b = -1.35$, yielding the unbalanced class proportion $P(Y = 1) = .1$. The other four cells with balanced class proportions had thresholds $a = 0.5$ and $b = -0.5$.

For generating outcomes with a logistic regression model, the rules were converted into linear terms z_j as follows:

$$Z_1 = R_1 = I(X_1 > a * X_2 > a);$$

$$Z_2 = R_2 = I(X_9 \leq b * X_4 \leq b);$$

$$Z_3 = R_3 = I(X_5 > a * X_{10} > a);$$

$$Z_4 = R_2 = I(X_7 \leq b * X_8 \leq b);$$

$$Z_5 = 1 - (\sum_{j=1}^{4} R_j).$$

The logit function, including an extra term with noise drawn from a standard distribution, to make the log-odds outcome for Y then becomes:

$$g(p_i) = \beta_1 Z_{i1} + \beta_2 Z_{i2} + \beta_3 Z_{i3} + \beta_4 Z_{i4} + \beta_5 Z_{i5} + \epsilon_i = \sum_{j=1}^{5} \beta_j Z_{ij} + \epsilon_i.$$

The link function $g(p)$ is the logit $g(p) = log\frac{p}{1-p}$, and $p = P(Y = 1)$. The inverse-logit function $g^{-1}(p)$ converts the log-odds resulting from the linear combination to probabilities p_i, yielding the outcomes Y_i:

$$Y_i = Binom\Big(\frac{exp(\sum_{j=1}^{5} \beta_j Z_{ij} + \epsilon_i)}{1 + exp(\sum_{j=1}^{5} \beta_j Z_{ij} + \epsilon_i)}, 1\Big) = Binom(p_i, 1)$$

Note that in this logistic regression model an extra term Z_5 was included (and the intercept was left out). Term Z_5, which was the complement of terms $Z_1, \ldots, Z_4$, had a negative weight (i.e., a very small probability of drawing 1) that accounted for predicting class 0. The error term followed a standard normal distribution ($\epsilon_i \sim N(0, 1)$) and also influenced the outcomes $g(p_i)$ on top of the indicator terms.

Threshold values for a and b, as well as weights for $\beta_1, \beta_2, \ldots, \beta_5$ were fixed beforehand for the eight models with Nagelkerke's $R^2 = 0.5$. The following combination of thresholds and weights yielded unbalanced class proportions for four of these cells: $a = 1.95$, $b = -1.95$, $\beta_1 = \beta_2 = \beta_3 = \beta_4 = 4$ and $\beta_5 = -4$. The final four cells had data drawn with this combination of thresholds and weights yielding balanced classes: $a = 0.9$, $b = -0.9$, $\beta_1 = \beta_2 = \beta_3 = \beta_4 = 3$ and $\beta_5 = -3$.

3.1.3) Methods and specification of method parameters

Six methods were applied in total: random forests, OTE, rule ensembles, and node harvest were among the applied methods. Two new methods were included in this section: rule ensembles, with a slightly different implementation in R than with RuleFit, and logistic regression.

In every replication of the simulation the optimal subset size S for random forests was optimized (with bootstrap sampling and chosen based on accuracy) with the caret package. As in Section 2.2.2, S was selected from candidate values 1, 3, 5, 8 and 10. The chosen S was also applied in fitting the OTE model within the same replication. All chosen values for optimal subset size were saved for further inspection of the influence of hyperparameter values in different design settings.

As with the current RuleFit interface it is not possible to save rules of the model in an R object, only the predictive and global recovery performance measures could be computed from models fitted with this package. The alternative package for fitting rule-based ensembles, PRE (Prediction Rule Ensembles; Fokkema

2016), was applied parallel to RuleFit to determine all performance measures. Differences between the two packages are that rule ensembles uses CART trees to extract rules from and PRE ensembles are based on conditional inference trees (Hothorn, Hornik, and Zeileis 2006). Furthermore, the regularization parameter in the former is determined through three-fold cross-validation and in the latter by ten-fold cross-validation. A secondary interest was to compare performances between these two slightly different rule-based ensemble implementations.

Parameters for node harvest and the two rule ensemble implementations were kept at default. Similar to RuleFit, PRE has the same default parameter values regarding subsample size and dependency parameter. All ensemble methods had comparable initial ensemble sizes, set to either 500 trees or approximately 1500 nodes (i.e., 500 two-way interaction trees with three leaves). Final ensemble size values were saved from the fitted OTE, node harvest, and rule ensemble (RuleFit and PRE) models to see whether ensemble sizes changed depending on the design factors applied in the simulation.

The final method was logistic regression, which was implemented as manipulation check for the Nagelkerke's R^2-based error design factor. Models were fitted with the following formulas for design cells with $error = .0$ and $error = .5$ respectively:

$$g(p_i)^{tree} = \alpha + \beta_1 Z_{i1} + \beta_2 Z_{i2} + \beta_3 Z_{i3} + \beta_4 Z_{i4} + \epsilon_i.$$

$$g(p_i)^{logis} = \alpha + \beta_1 Z_{i1} + \beta_2 Z_{i2} + \beta_3 Z_{i3} + \beta_4 Z_{i4} + \beta_5 Z_{i5} + \epsilon_i.$$

3.1.4) Measures of recovery performance

All methods were assessed on interpretability by their ability to recover the true effects present in the data. Recovery performance was again assessed with three categories: global recovery performance, specific recovery performance regarding correct variable interactions, and regarding split points.

The specific recovery performance measures for variable interaction and split points specifically could only be

calculated for node harvest, rule ensembles made with PRE, and logistic regression. For node harvest and rule ensembles these measures were deduced from the ten most important ensemble members (sorted on weight or importance values). Of these ten ensemble members, the amount of times a correct interaction was found was counted. The split points of variables were only saved conditional on whether the correct interaction was found (e.g., when the interaction $x_1\&x_2$ was found in one of these ten ensemble members, the split points of x_1 and x_2 from that ensemble member were saved). Specific recovery performance regarding splitting values was then determined as the rate at which the found split point value lied within $a \pm 0.5$ or $b \pm 0.5$ [1]. Note that in the eventual simulation extraction of the split points from PRE ensemble members failed, hence we will not further discuss this. For logistic regression specific recovery performance was based on the linear coefficients for the interaction terms and their significance. Of every fitted model, the fitted values of the interaction terms were assessed and an indicator reflected whether the all effects of these interaction terms were significant ($p \leq \alpha = .10$).

For global recovery performance, the order of variable importances of every model fitted were used. When the values were sorted, an indicator was computed that reflected whether variables x_3 and x_6 (and only these) received the lowest variable importance values (i.e., the model correctly detected the most and least important variables involved in predicting responses). The indicator also reflected whether at least all the eight involved variables were included in the final ensembles. Such an indicator was made for every replicate within each design cell. For logistic regression global recovery performance could not be determined, as the linear terms were solely based on interactions between variables.

3.1.5) Measures of model performance and analyses thereof

Accuracy, Brier scores, AUC values, and Press' Q values were computed in every replication for every cell for all five methods. Additionally, Nagelkerke's R^2 were computed from the fitted logistic regression models. Nagelkerke's R^2 (Nagelkerke 1991) is equivalent to R^2 for linear models, but is used for generalized linear models. Nagelkerke's R^2 is calculated with the number of binary observations, with the maximized likelihood

[1] 0.5 is the same as $\pm 0.5\sigma$ within a population of Z variables, which could be applied here as variables were drawn from a multivariate normal distribution with variances of 1

(or deviance) under the current model and the maximized likelihood/deviance under the null distribution:

$$R^2 = \frac{1 - (\hat{L}_0/\hat{L})^{2/n}}{1 - \hat{L_0}^{2/n}} = 1 - \frac{exp((D - D_{null})/n}{1 - exp(-D_{null}/n)}$$

(Faraway 2006). Calculation of this measure was embedded in every replicate to check whether the correct amount of signal (or error) allowed in the model was indeed achieved.

Computed performances of every method per replicate were collected in a final dataset for analysis. This performance dataset had the following structure: the rows were the subjects, with one subject representing a simulation dataset (or replicate), and the columns were the design factors. Five repeated measures were done on every subject: every method counted as a repeat, yielding a total of five responses. Every method had a separate column for their respective responses. A repeated measures ANOVA was applied to determine whether differences in performances between methods and the other design factors involved were statistically significant. The within-subjects factor specified was the classification method applied; the between-subjects factors were the three design factors.

The significance level of $\alpha = .05$ was applied. Although significant p values are indicators of potentially important effects, with such large simulation studies effect sizes give better indications of the magnitude of factor contributions in influencing the changes in outcomes (Sullivan and Feinn 2012). The effect size for repeated measures ANOVA is η^2, which is the proportion of variance explained by a factor. Factors with $\eta^2 \geq .06$ (medium effect size; Cohen 1988) only were considered. If a term had an effect size $\eta^2 \geq .06$, the within-subject contrasts (i.e., contrasts between methods) were consulted to determine sources of substantial differences between random forests (as benchmark) and any other method. Contrasts were chosen based on partial η^2, which is the effect size measure for one effect, after correcting for the other effects (Richardson 2011). As we indeed were interested in comparing every method separately with random forests as benchmark, partial η^2 instead of η^2 is used here to determine the effect of a particular contrast on its own. Again only contrasts with partial $\eta^2 \geq .06$ were considered.

The residuals of the initial ANOVA analyses showed that there were some outliers for every outcome measure per method (corresponding to extremely bad model performance values; see histograms in Appendix D, Fig. 1). These were mainly the same observations (Appendix D, Fig. 2-4). This could have been the consequence of a bad seed in data generation. Hence all observations where logistic regression accuracy rates were <0.75 were left out of the analysis as an extra precaution. For design cells where the average accuracy rates were ≈ 0.7 (cell factor levels $error = .5$ and $P(Y = 1) = .5$) observations with accuracy rates <0.5 were excluded.

3.1.6) Software

Simulations were performed in R version 3.3.1. Additional to previously specified R packages, the PRE package (Prediction Rule Ensembles; Fokkema 2016) was also used here. R scripts of the main functions used for generating data and performing simulations are given in Appendix C. The ANOVA analyses were performed in IBM SPSS version 23 (SPSS and others 2015).

3.2) Results

As interpretability is the main concern, first specific recovery performance results are summarized, after which global recovery performance of all methods is described. Then model performance assessments are summarized. This section is concluded with a check of simulation manipulations by assessing logistic regression performances, to determine whether the simulation went according to plan.

Regarding training during the simulation, random forest model selection resulted in a choice of generally larger random subset size S values ($S \geq M/2$) for cells where more signal was present in datasets (i.e., $1 - error > .5$; Appendix D, Table 7). The final ensembles sizes of OTE and node harvest did not show large deviations between varying design factors (Appendix D, Table 8). RuleFit however often had much larger final ensemble sizes when more signal was present in the data, especially with unbalanced classes (i.e., $P(Y = 1) = .1$). PRE mainly seemed to have increased ensemble sizes when training set sizes were very large ($n_{train} = 5000$).

3.2.1) Specific recovery performance

In node harvest, the average rate of finding the correct interactions between variables increased with increasing training set size, $P(Y = 1)$, and/or signal (see Table 6). The conditional split point recovery for node harvest increased likewise. Conditional split point recovery was either reasonable or excellent, even when true variable interaction recovery was low.

For rule ensembles with PRE, true interaction recovery was either higher than node harvest (with balanced classes and $e = 0$) or much lower in the noisier settings (Table 6), despite the high variable recovery performances. In the cells where $e = .5$ and unbalanced class outcomes, there was almost no recovery of true variable interactions.

Logistic regression specific recovery performance rates were often very low (Table 6). It often happened that no significant effect of these indicator variables were found: no proper solution for the coefficients of the linear terms could be found or the standard errors got enormous values. This caused the p values of these indicator terms to be ≈ 1.

Table 6: Specific recovery performance rates for node harvest (NH), rule ensembles with PRE implementation and logistic regression. Rates for NH and PRE are based on the 10 most important ensemble members from the 100 models fitted per design cell, for logistic regression they are based on how often all interaction effects were found to be significant ($\leq \alpha = .1$). Split point recovery rates are only given for node harvest.

$P(Y=1)$	$1-error$	n_{train}	NH True Interaction Rate	NH True Split Rate	PRE True Interaction Rate	Logistic regression True Interaction rate
.1	.5	250	.09	.56	.04	.10
		500	.11	.65	.04	.09
		1000	.17	.69	.03	.04
		5000	.58	.95	.02	.48
	1.0	250	.66	.99	.05	.00
		500	.75	.99	.48	.00
		1000	.68	.99	.56	.00
		5000	.68	1.00	.70	.00
.5	.5	250	.23	.84	.02	.64
		500	.45	.91	.09	.91
		1000	.57	.98	.26	.99
		5000	.60	1.00	.29	1.00
	1.0	250	.56	1.00	.67	.00
		500	.59	1.00	.80	.00
		1000	.61	1.00	.85	.00
		5000	.68	1.00	.99	.00

Table 7: Global recovery performance rate per tree-based method per design cell.

			n_{train}			
$P(Y = 1)$	$1 - error$	Method	250	500	1000	5000
.1	.5	Random Forests	.40	.83	.99	1.00
		OTE	.39	.64	.94	1.00
		Node Harvest	.05	.13	.14	.74
		Rule Ensembles (RuleFit)	.20	.73	.88	1.00
		Rule Ensembles (PRE)	.50	.63	.81	1.00
	1.0	Random Forests	.72	.77	.84	.80
		OTE	.71	.80	.83	.79
		Node Harvest	.47	.61	.79	.77
		Rule Ensembles (RuleFit)	.75	.75	.80	.85
		Rule Ensembles (PRE)	.71	.93	.99	1.00
.5	.5	Random Forests	.70	1.00	1.00	1.00
		OTE	.38	.91	.99	1.00
		Node Harvest	.22	.66	.95	1.00
		Rule Ensembles (RuleFit)	.58	.88	.99	1.00
		Rule Ensembles (PRE)	.86	.82	.85	.86
	1.0	Random Forests	.88	.87	.88	.78
		OTE	.89	.88	.85	.81
		Node Harvest	.77	.84	.80	.77
		Rule Ensembles (RuleFit)	.86	.87	.84	.77
		Rule Ensembles (PRE)	.92	.88	.90	.81

3.2.2) Global recovery performance

Overall, global recovery performance improved when training set size increased (Table 7), especially in the noisier settings; this was similar to changes observed in specific recovery performance rates. Furthermore, rates were higher when more signal (less error) was present in the data. Within the error factors, recovery performances were higher with unbalanced class outcomes and $n_{train} > 500$; rates were higher with balanced class outcomes and with $n_{train} < 1000$. In settings with full signal, recovery performance rates of all methods were comparable. Random forests had even better variable recovery in noisier settings than in settings with more signal in the data.

Of all methods, random forests and rule ensembles with PRE most frequently had the highest global recovery performance rate. Node harvest had very low global recovery performances in settings with more noise and/or lower training set sizes, despite that its specific recovery performances were still reasonable in such settings.

Table 8: Summary of most important within-subject effects ($\eta^2 \geq .06$) from the simulation per performance measure, including corresponding method contrasts with the benchmark random forests (partial $\eta^2 \geq .06$).

Measure	Effect	η^2	Contrasts:	OTE	NH	RuleFit	PRE
Accuracy rate	$Method$	.285			x	x	
	$Method * error$	.133		x	x	x	x
	$Method * n_{train} * P(Y = 1)$	.106			x		x
Brier	$Method$	.439			x	x	x
	$Method * error$	.124		x		x	x
	$Method * P(Y = 1)$	.157			x	x	x
	$Method * error * P(Y = 1)$	.063			x	x	x
AUC value	$Method$	.198		x	x	x	x
	$Method * n_{train}$	.074			x	x	x

3.2.3) Predictive performance

All within-subject effects for all model performance outcomes were significant (Greenhouse-Geisser corrected p values <.05), but here we focus solely on effects with $\eta^2 \geq .06$. Table 8 summarizes the most important within-subject effects and corresponding contrasts between random forests and the other methods (see Appendix D, Tables 9-11 for more detailed summaries of average model performances); Figure 6 contains plots corresponding to interactions of substantial size. The main effect of *method* even had a large effect size ($\eta^2 \geq .14$; Cohen, 1988) in every model performance measure (Table 8), already confirming that there are differences in classification performance of the tree ensemble methods, regardless of the design factors.

In the following paragraphs the most important effects contributing to differences in model performances are explained per measure. Note that the assumption of residual normality was violated despite the removal of some outliers, but most distributions remained approximately symmetric within the quantiles (Appendix D, Figs. 2-4).

The within-subject effects that explained most of the variance of average accuracy, were the main effect of *method*, the interaction $method * error$ and $method * proportion * n_{train}$. For *method*, the largest contrast was with node harvest (Table 8), which had lower rates than random forests at almost every point (Figs. 6a1, 6a2; Appendix D, Table 9). The $method * error$ interaction in Fig.6a1 is visible: with less error (or more signal) accuracies increase on average for all methods, but with varying strenghths. All methods had substantial contrasts with random forests on this effect. In the noisier setting random forests had slightly

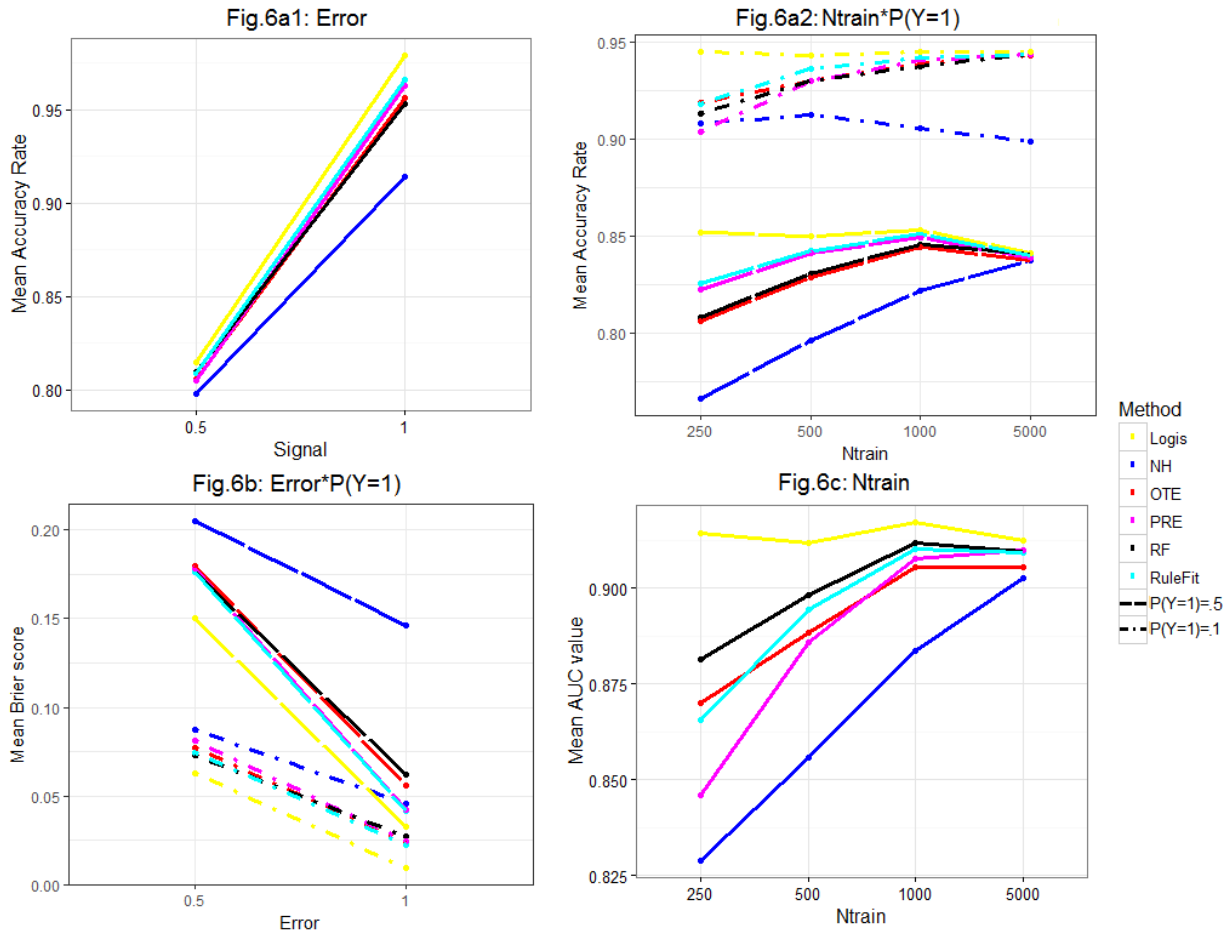


Figure 6: Average performance measure per method (RF=random forests, OTE=optimal trees ensembles, NH=node harvest, RuleFit=rule ensembles with RuleFit, PRE=rule ensembles with PRE) plotted for every interaction with the *method* effect ($\eta^2 \geq .06$) mentioned in Table 6.

higher accuracy rates than the other methods and in the setting with no error this effect was opposite, with the exception of node harvest. For the final interaction effect $method * n_{train} * proportion$ (Fig. 6a2), the greatest contrasts were with node harvest and PRE. For node harvest the average rates did not increase when n_{train} increased and when $P(Y = 1) = .1$, unlike the other methods. The contrast with random forests is also visible in the plot when $P(Y = 1) = .5$ and $n_{train} = 5000$: here node harvest catched up with the other methods, while for lower n_{train} it had much lower accuracy rate values. For PRE the contrasting interaction is mainly visible where $P(Y = 1) = .1$ and n_{train} increased from 250 (lower average rates) to 500 (higher rates); with larger n_{train} it continued to perform similarly as random forests.

$Method$, $method * error$, $method * P(Y = 1)$ and $method * error * P(Y = 1)$ had the greatest influence on changes in Brier scores (Table 8). Only the three-way effect is discussed, as it includes both design factors of the two-way effects. For the *method* and the tree-way effect, the largest contrasts were with all methods except OTE: OTE had average Brier scores similar to those of random forests. The three-way interaction showed a drastic decrease of Brier scores in cells with unbalanced class outcomes and decreased amount of noise (Fig.6b). This effect was the least severe for node harvest, but the strongest for both rule ensemble methods. The change in average Brier scores for varying amounts of noise was more severe when class outcome proportions changed. For both $P(Y = 1)$ values, the rule ensemble methods had lower average Brier scores than random forests when more signal was present in the data.

In cells with unbalanced class outcomes (i.e., $P(Y = 1) = .1$) the average accuracies for all methods were $\approx P(Y = 0) = .9$ and Brier scores were very low. This is noteworthy, as when one of the classes has the upperhand, it is easier to achieve higher accuracy rates when predicted outcomes mainly have the labels of the dominant class.

The greatest effects affecting changes in AUC values were *method* and $method * n_{train}$. On the *method* main effect, the most important contrasts of random forests were with all methods (Table 8). Only OTE did not contrast substantially with random forests on the $method * n_{train}$ interaction, the changes in average AUC values were similar between these two methods (see parallel slopes in Fig. 6c). For lower training set sizes the rule ensemble models had lower AUC values, but had stronger increases in AUC values when training set sizes

increased as well (Fig. 6c). For very large training set sizes the average AUC values were similar to those of random forests. Node harvest showed increasing AUC values when training set size increased, although these changes were less extreme. Again, node harvest had the lowest average AUC values in all cases.

Press' Q values were significant (i.e., $> \chi^2_{.05}(1) = 3.84$) for every method in every design cell in at least 99% of the cases. Hence almost all fitted models predicted classes better than chance.

3.2.4) Manipulation check: classification with Logistic Regression

The Nagelkerke's R^2 values from the fitted logistic regression models were very close to or equal to the $1 - error$ allowed in a design cell (Appendix D, Table 6). This confirmed that data generation went according to the specified underlying logistic regression or tree models.

Logistic regression had the best (average) performances (Figure 6; Appendix D, Tables 9-11). This was as expected, since this method was the manipulation check and fitted models had formulas specified corresponding to the true underlying models. Still, it was not the perfect predictor, despite that the models were fitted with coefficients representing the true interactions present in the training data. One explanation could be that the true outcomes were constructed with some additional error ϵ. Another reason could be that some rules overlapped in producing outcomes, which made it difficult to separate effects. A final explanation is that the coefficient values fitted did not exactly correspond to the coefficients specified for the true underlying models (this also relates to the problem mentioned in Section 3.2.1 regarding the high standard error solutions for the coefficients of the fitted models).

4) Application to a real dataset

The application concerned patients with Alzheimer's disease and controls. The dataset used here was kindly provided by M. Bouts and C. Möller (Faculty of Social Sciences, Leiden University, the Netherlands).

4.1) Background information

Alzheimer's disease is the most common form of dementia, especially in patients $\geq$ 65 years (Kester and Scheltens 2009). Alzheimer's disease can be identified through certain psychological tests or assessment of clinical criteria, by making a magnetic resonance imaging (MRI) scan of a patient's brain or by autopsy (Möller et al. 2015). With MRI scans, damage or grey matter and white matter atrophy in the brain can be revealed.

The dataset of Bouts & Möller had 65 subjects that were either classified as control/healthy (0, n=35) or having Alzheimer (1, n=30).

The dataset had in total 173 variables. Three demographic variables were included: age, gender (37 patients were 1=male, 28 2=female) and treatment center (41 patients from 1=VUMC Alzheimercentrum Amsterdam, 24 from 2=Erasmus MC Alzheimercentrum Rotterdam). Ages from the patients ran from 51 to 78, with the rounded mean age being 64. The remaining part of the features were based on voxel measurements or ratios. The features were roughly divided into the following groups: measurements of grey matter density (GMD; regarding brain grey matter atrophy), white matter density (WMD; regarding integrity of white matter), mean diffusivity (MD; mapping the diffusion process of water molecules within white matter areas) and fractional anisotropy (FA; measuring the direction of water flow in white matter areas). Such characteristics of grey and white matter are measured in various parts of the brain. GMD and WMD values are extracted from voxels of MRI images. FA and MD is determined with diffusion tensor imaging (DTI) and are ratios or proportions. FA is the fraction of longitudinal and oblique water flow in white matter tracts and MD is the directionally averaged rate of diffusion (Head et al. 2004; Versace et al. 2008).

4.2) Workflow of application and evaluation

Before assessing model performance on this dataset, relevant (hyper)parameters were selected or optimized through a cross-validation procedure. Variable involvement (variable importances and, where possible, specific information recovery) was assessed by fitting a single final model to the entire dataset. Predictive performances were assessed through a separate cross-validation procedure rather than training and testing, as the dataset contained a limited number of subjects.

4.2.1) Parameter specifications of the methods

(Initial) Ensemble sizes were set to 500 trees for random forests and OTE. Maximum interaction order for node harvest was set to 2; equivalently, the average tree size for rule ensembles was set to 4. The maximum number of initially generated nodes and rules were hence set to 2000. For all methods (except rule ensembles) node size was set to 5.

For random forests, the random subset size hyperparameter S was fine tuned to the entire dataset by separate ten-fold cross-validation. This was repeated 50 times to achieve a more stable outcome. The following values for S were attempted: 1, 3, $\lfloor {}^2log(M+1) \rfloor = 7$, $3^2 = 9$, $\lfloor \sqrt{M} \rfloor = \lfloor \sqrt{173} \rfloor = 13$, $3^3 = 81$, $173/3 = 58$, $173/2 = 87$ and 173 (similar to Section 2.2.2). The optimal value was $S = 9$ (average accuracy rate .863). This S was applied in the model assessment procedure for random forests and OTE.

For OTE, the percentage of tree candidates for the final OTE ensemble was chosen through 50 times repeated ten-fold cross-validation, with candidates 20%, 15%, 10%, 5% and 1%. Wilcoxon signed rank tests determined that only the paired differences between 20% and 15% regarding accuracy values were not significant ($p_{adj}^{Bonf} = .266$), hence the percentage of 15% was applied to produce slightly smaller ensembles.

The tuning parameters of rule ensembles and node harvest were set to defaults, as specified in Sections 2.3.2 and 2.4.2 respectively.

4.2.2) Interpretation

From the final models fitted (one per method), variable importances and other output relevant for interpretation were inspected. The most important predictors and, where possible, prediction rules, and corresponding threshold values related to predicting Alzheimer´s disease are shortly described.

4.2.3) Measures of model performance

The model assessment procedure was done through ten-fold cross-validation, which was repeated 50 times to gain a distribution of all model performance values for every method (see Appendix E for code). Every repeat was treated as an observation. The cut-off value of class probabilities applied was .5. As the dataset had 30 class 1 cases and 35 class 0 cases, the proportion of Alzheimer's patients was 30/65=.462, .5 was a reasonable cut point. Additional to previously used performance measures, specificity, sensitivity and corresponding cut points were computed. From specificity the first values $\geq .95$ and $\geq .90$, corresponding sensitivity values, and cut points were averaged across all repeats. Conversely, the first values of sensitivity that were $\geq .95$ and $\geq .90$, were saved along with paired averaged specificity and averaged cut point values.

Shapiro-Wilk normality tests confirmed that not all performance measure distributions of every method were approximately normally distributed ($p < .05$ for at least one method per performance measure). Hence, differences in performances were determined through nonparametric tests (as applied in e.g., Maroco et al. 2011). Wilcoxon signed rank tests were applied to determine differences in prediction performances between random forests as benchmark and the other classifiers, where p values were adjusted with a Bonferroni correction. An overall significance level of $\alpha = .05$ was used.

4.3) Results

4.3.1) Model interpretation

4.3.1.1) Global recovery performance

The ensemble sizes of the final models were 500 for random forests, 43 for OTE, 23 for rule ensembles, and 68 for node harvest. In all of these models parts of the hippocampus were selected as important predictors, as well as parts of the temporal gyrus (Table 9). All models (mainly) selected grey matter density predictors, only node harvest selected a white matter density predictor and an MD predictor. It appeared that changes in grey matter density of various parts of the brain were important indicators of Alzheimer's disease. This is in accordance with other studies on distinguishing Alzheimer's disease (such as Möller et al. 2015). Grey matter areas with substantial decrease in mass that are mentioned in both Table 9 and Möller et al. (2015), either by their own findings or by previous studies, are the superior and middle temporal gyrus, the parahippocampal gyrus and the hippocampus.

Apart from random forests, where naturally all variables are used at least once, the other models did not contain all of the predictors in the dataset (i.e., their variable importance values equal 0). The OTE model excluded 67 of the 173 predictors, the rule ensembles model excluded 163 variables, and the node harvest model excluded 116.

Random forests and OTE had very low importance values for all variables, indicating that the variables did not appear often in their decision trees. This could be explained by the fact that trees were constructed with a high dimensional dataset with many variables, from which only a small fraction was selected at random in every next split. Also, in different folds different sets of observations were used which might have contained completely different sets of variables and outcomes. Node harvest and rule ensemble variable importances were relative measures (computed as described in Section 2), making it difficult to determine the impact of every variable on the ensemble model such as in the forest models.

Table 9: The ten most important variables selected by the four final models fitted to the Alzheimer dataset with corresponding variable importance values (sorted on mean decrease Gini index $\overline{\Delta(G)}$ or relative proportions $\hat{p}$). All variables are GMD measurements of various brain areas, except for three variables selected from the OTE and node harvest model: these are changes in WMD(*) and MD(**) respectively. The abbreviations represent divisions (d) of brain parts, such as anterior/posterior/superior (a/p/s) and left/right (l/r).

Random Forests	$\overline{\Delta(G)}$	OTE	$\overline{\Delta(G)}$	Node Harvest	$(\hat{p})$	Rule Ensembles	$(\hat{p})$
Cingulate gyrus pdr	1.1	Left hippocampus	2.5	Angular gyrus l	1.00	Right hippocampus	1.00
Frontal medial cortex r	0.9	Cingulate gyrus pdr	2.5	Middle temporal gyrus pdr	.99	Lateral occipital cortex sdl	.83
Left hippocampus	0.9	Lateral occipital cortex sdl	2.2	Middle temporal gyrus adl	.98	Left thalamus	.83
Parahippocampal gyrus pdl	0.9	Frontal medial cortex r	1.2	Paracingulate gyrus l	.92	Cingulate gyrus pdr	.56
Right hippocampus	0.8	Temporal pole right	1.0	Occipital fusiform gyrus l	.85	Angular gyrus l	.56
Paracingulate gyrus l	0.7	Middle temporal gyrus pdl	0.9	Parahippocampal gyrus pdr	.82	Inferior temporal gyrus pdl	.55
Lateral occipital cortex sdl	0.7	Paracingulate gyrus l	0.9	Superior temporal gyrus pdl	.79	Cuneal cortex l	.25
Cingulate gyrus pdl	0.7	Cingulate gyrus adr	0.8	Left hippocampus	.75	Frontal operculum cortex r	.19
Cingulate gyrus adr	0.6	Parahippocampal gyrus pdr	0.8	Cingulum hippocampus l*	.67	Precuneous cortex l	.17
Superior temporal gyrus pdl	0.6	Precuneous cortex l	0.8	Tract0009 forceps minor**	.62	Middle temporal gyrus adr	.14

Table 10: Ten nodes with the highest weights selected from the final node harvest ensemble on the Alzheimer dataset. Per node the variable interactions and corresponding split points, node weight $\mathbf{w}_j$, training sample size n_j, and predicted average value $\hat{y}_j$ per node are given. All variables mentioned are part of the GMD variable group.

Node q_j	Rule	$\hat{y}_j$	n_j	$\mathbf{w}_j$
q_{14}	Parahippocampal gyrus pdr $>$ 1320 & Angular gyrus l $\leq$ 3540	1	5	.450
q_{51}	Superior temporal gyrus pdl $\leq$ 3520 & Left hippocampus $\leq$ 3730	1	21	.372
q_{25}	Paracingulate gyrus l $>$ 5290 & Cingulate gyrus pdr $\leq$ 4530	1	6	.307
q_{15}	Cingulate gyrus pdr $>$ 4520 & Occipital pole l $\leq$ 5750	1	5	.265
q_{16}	Cingulate gyrus pdr $\leq$ 5240 & Occipital fusiform gyrus l $>$ 4370	.2	5	.239
q_{21}	Cingulate gyrus pdr $\leq$ 4530 & Middle temporal gyrus adl $\leq$ 1500	1	6	.188
q_{28}	Cingulate gyrus pdr $>$ 4540 & Left thalamus $\leq$ 2670	1	6	.183
q_{8}	Middle temporal gyrus pdr $>$ 5030 & Left hippocampus $\leq$ 2900	1	5	.180
q_{47}	Middle temporal gyrus pdr $\leq$ 4420	1	15	.170
q_{64}	Inferior temporal gyrus pdl $\leq$ 4010 & Left hippocampus $\leq$ 3030	.033	30	.154

4.3.1.2) Specific recovery performance

The ten nodes with highest weights from the final node harvest model are displayed in Table 10. Node q_{47} concerned a main effect, the other nodes in this selection concerned two-way interactions. Variables such as the hippocampus, cingulate gyrus and temporal gyrus appeared most often in these nodes. Most nodes contained very few observations ($n_j < 10$). Eight of the ten nodes predicted Alzheimer's disease (i.e., the predicted value was 1). For some of these nodes, the accompanying thresholds indeed indicate that grey matter mass below a certain value in certain parts of the brain is an indicator of this disease. However, some thresholds would also predict Alzheimer's disease if GMD of certain brain parts would be above certain volumes (e.g., nodes q_{14} and q_{25}.

Split points of variables chosen more than once were often in agreement with each other, as their values were similar (such as the split points of the cingulate gyrus *pdr* and left hippocampus; Table 10). The middle temporal gyrus was also selected three times, but this regarded two different parts (*adl* and *pdr*), which is why the three corresponding split points are different.

Table 11: First five most important rules of the final rule ensemble model fitted to the Alzheimer dataset, with support values s, coefficient values a, and relative importance proportions I. All predictors are part of the group of GMD variables.

Rule order	Rule	Support s_j	a_j	I_j
1	Inferior temporal gyrus pdl > 3750 & Angular gyrus l > 3545 & Cingulate gyrus pdr > 4538	.51	-1.508	1.00
2	Lateral occipital cortex sdl > 13030 & Left thalamus > 2671 & Right hippocampus > 3170	.57	-1.090	.72
3	Lateral occipital cortex sdl > 13030 & Cuneal cortex l > 1666 & Left thalamus > 2671 & Right hippocampus > 3170	.49	-0.918	.61
4	Lateral occipital cortex sdl > 13030 & Frontal operculum cortex r $\leq$ 1816 & Left thalamus > 2671 & Right hippocampus > 3170	.48	-0.741	.49
5	Middle temporal gyrus adr > 1444 & Precuneous cortex l > 8740 & Right hippocampus > 3170	.48	-0.388	.26

The first five most important rules of the final rule ensemble are summarized in Table 11. The top five is presented instead of the top ten, as from the sixth rule onward all relative rule importance values were below .05 (i.e., 5%). All five rules shown had negative coefficients, so satisfying one of these rules indicated a lower probability of being an Alzheimer patient (i.e., a higher probability of being a control case). The support values show that for every rule almost half of the data was used to construct them. All variables included in these rules are also displayed in the variable importance table (Table 9). Within these rules, some variables were even selected more than once with the same split points. The split points of the variables generally state that when GMD is above a certain threshold, the person has a higher probability to belong to the control group. This coincides with the general diagnostics of Alzheimer patients where these densities are shrunken compared to controls. Note that all displayed rules are three-way interactions, despite that average tree size $\bar{U} = 4$ (i.e., mainly two-way interactions) was specified.

Some variables chosen by these rules overlapped with the variables from the node harvest model (Table 11 and 10 respectively). These were the angular gyrus *l*, the cingulate gyrus *pdr*, the left thalamus and middle temporal gyrus *adr*. The corresponding split points of both models of these variables were also in agreement with each other as they had similar values.

Table 12: Median model performance measure values for every method, taken over all 50 cross-validation repetitions. 95% confidence intervals ($CI_{.95}$) for the AUC values is based on average AUC values.

Method	Accuracy	Brier score	AUC value	AUC $CI_{.95}^{lower}$	AUC $CI_{.95}^{upper}$	Press'Q
Random Forests	.862	0.112	.940	.937	.941	33.98
OTE	.846	0.119	.916	.915	.924	31.15
Node Harvest	.831	0.130	.896	.883	.897	28.45
Rule Ensembles	.769	0.172	.825	.817	.840	18.85

4.3.2) Cross-validated model performances

The random forest values were overall the best: the Brier scores had the lowest median and the other measures the highest medians (Figure 7, Table 12). For all measures, OTE came second to random forests, node harvest third, and rule ensembles was surprisingly the worst performer. Wilcoxon signed rank tests demonstrated that between random forests and every other method the differences in all performance measure values were significant (all $p_{adj} < .05$). Yet every method classified better than chance (all Press'Q values ≥ 3.84; see Figure 7d).

Fig. 8 shows the averaged ROC curves per method: on almost every point, random forests (black) had values closest to the upper left corner and thus was the superior discriminator.

4.3.2.1) Sensitivity and specificity

Within every fixed value for specificity and sensitivity, the corresponding sensitivity or specificity values differed significantly between random forests and the other three methods (all Bonferroni-corrected $p_{adj} < .05$). The only exception was where sensitivity was fixed at .90: here OTE had specificity values comparable to random forests ($p_{adj} = 1$) and the corresponding average cutpoints were also highly similar. Yet in every situation random forests had the highest sensitivity values paired with fixed specificity values and highest specificity values paired with fixed sensitivity values (Table 13).

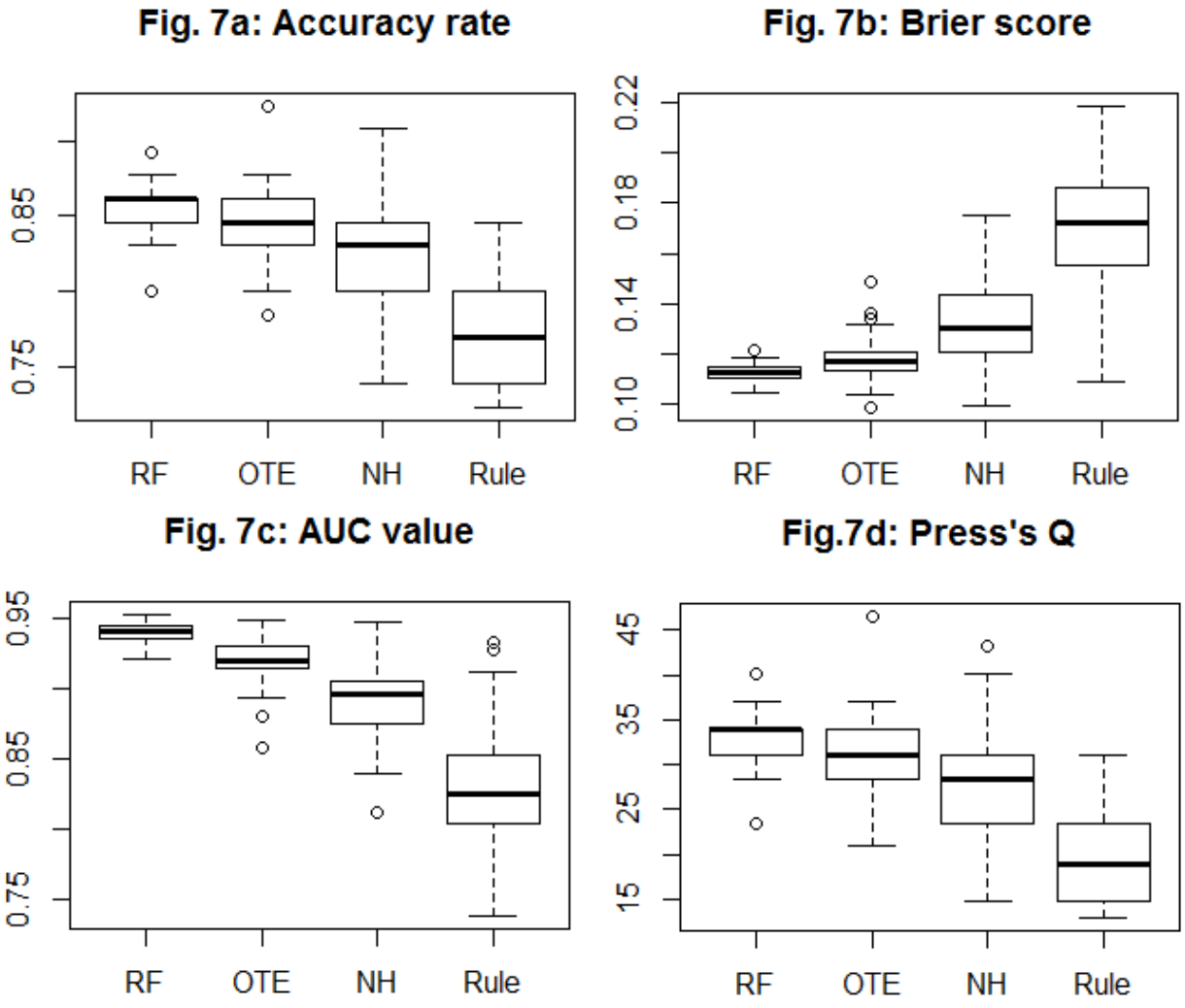


Figure 7: Boxplots of the distributions of model performances from the repeated cross-validation procedure for the Accuracy rates, Brier scores, AUC values and Press' Q values per method (RF=random forests, OTE=optimal trees ensembles, NH=node harvest, Rule=rule ensembles). Random forests performed best compared to each other method.

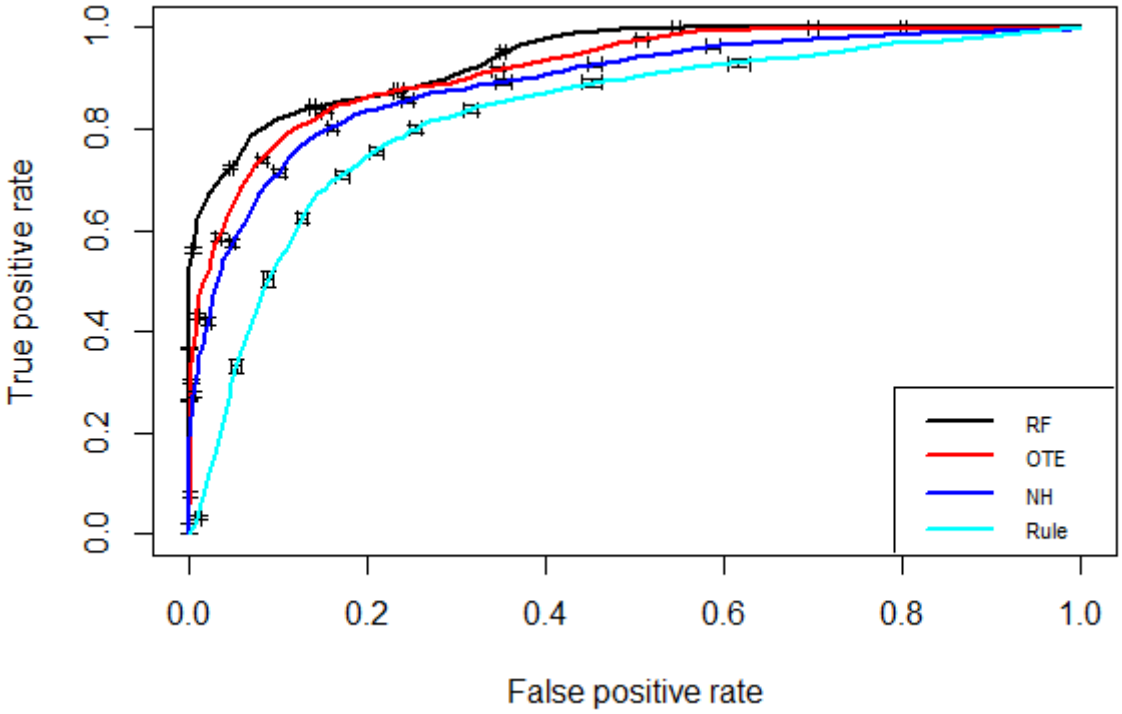


Figure 8: Averaged ROC curves of all cross-validation repeats per method (RF=random forests, OTE=optimal trees ensembles, NH=node harvest, Rule=rule ensembles), with standard error bars. Averaging was done based on thresholds (i.e., cutpoints), as described in Fawcett (2006). The standard error bars are plotted at the threshold points over which was averaged.

Table 13: Averaged sensitivity with fixed specificity values (left) and averaged specificity with fixed sensitivity values (right), with corresponding cutpoints for classification. First the minimum fixed value is given (for both sensitivity and specificity), then the average of the first specificity values above this value are given together with the corresponding averaged sensitivity value and averaged cutpoint. This is also done for fixed sensitivity and corresponding specificity and cutpoint values. Note that most true specificity and sensitivity values at the cutpoints were almost never exactly .90 or .95.

Method	Specificity (fixed)	Sensitivity	Cutpoint	Sensitivity (fixed)	Specificity	Cutpoint
Minimum fixed value: .90						
Random Forests	.91	.83	.52	.90	.74	.35
OTE	.91	.77	.56	.90	.71	.35
Node Harvest	.91	.73	.58	.91	.70	.32
Rule Ensembles	.91	.57	.74	.90	.57	.21
Minimum fixed value: .95						
Random Forests	.97	.70	.60	.97	.66	.28
OTE	.97	.67	.65	.97	.57	.21
Node Harvest	.97	.57	.71	.97	.46	.11
Rule Ensembles	.97	.30	.92	.97	.31	.05

5) Discussion

An alternative tree ensemble method with better interpretability than random forests was sought in this study. Among the candidates were optimal trees ensembles (OTE), node harvest, and rule ensembles. Suitability of being such an alternative to random forests was determined by specific recovery performance (regarding variable interactions and split points; to measure correctness of prediction rules), global recovery performance (main variable importances), and predictive performance. This was done through a simulation study, embedded with varying situations, and an application to a high-dimensional MRI dataset. Random forests' lack of interpretability but very good prediction performance were taken as baseline. Our results showed that suitable interpretational alternatives to random forests were node harvest and rule ensembles, but their information recovery performance and predictive performances highly depended on the type of data.

Interpretability in this study could only be investigated if it was possible to assess specific recovery performances. Specific recovery ability was the most important assessment for this study and regarded correctness of variable interactions and variable thresholds contained in prediction rules. It could only be assessed for node harvest and rule ensembles (PRE in the simulation, RuleFit in the application): these are two types of methods that produce relatively small final ensembles with simple ensemble members. The simplicity of these members is a characteristic that makes these methods highly suitable for interpretational purposes. They give insight in variable interactions and thresholds involved in predicting certain class outcomes. This is exactly why it was important to assess specific recovery performance: the information contained in the (most important) ensemble members should be correct, otherwise application of these methods could lead to incorrect interpretations. On the contrary, OTE has highly complex ensemble members and only global recovery performance could be assessed for this method, hence it has no added value compared to random forests regarding interpretation.

5.1) Discussion of results

In the simulation study, specific recovery performance of rule ensembles with PRE was worse in mainly settings with unbalanced class outcomes. Node harvest on the contrary had reasonable or good specific

recovery performance in such settings. It had perfect conditional split point recovery in design cells with unbalanced classes and no noise (Nagelkerke's $R^2 = 1$). Recovery performance in unbalanced class settings for PRE (or rule ensembles in general) could have been harmed by the presence of the dominating class 0, resulting in less support for rules predicting 1. Noisier settings of the simulation resulted in poorer interaction recovery performance for node harvest and PRE. The effect of noisiness on these outcomes was also observed in the application study. There the RuleFit procedure produced more complex ensemble members (i.e., complexer interactions) than the average tree depth specified, and no agreement in variable interactions between the final RuleFit and node harvest ensemble was observed. Upon closer inspection, the rules produced by RuleFit showed a quick decline in relative importance, which could have been due to the limited support available for individual rules from this small sample size dataset.

Conditional split point recovery of node harvest on the other hand was still relatively well or excellent in noisier settings of the simulation. Although true split point recovery could not be determined within the application study on the MRI dataset, there was an agreement in certain variable split points in the final node harvest and RuleFit ensemble. Agreement of ensemble members within or between ensembles indicates that these split points could indeed be important in discriminating outcomes. The toy data example demonstrated node harvest's ability to correctly retrieve the true thresholds, although this was often paired with incorrect variable interactions. All results indicate that node harvest is able to recover important variable thresholds in various types of data.

Aside from the importance of specific recovery performance, global recovery performance was also of great interest. Although variable importances do not give detailed insights in how variables are involved in predicting outcomes, and hence are less suitable for interpretational purposes, they do give insight in basic involvement of variables in predictions. Its assessment indicated whether the correct variables were included in the ensemble in the first place. Moreover, this measure could be computed for every method incorporated in this study, allowing for equal comparisons.

In the simulation study, random forests had among the best global recovery performance rates, which were even better in noisier settings. OTE had comparable variable recovery performance rates, indicating that with

smaller random forests ensembles similar information on important variables in the data can be extracted. Node harvest had the worst variable importance recovery in many settings: only with an enormous amount of training data and/or data with a lot of signal node harvest was able to properly assess variable importance. Peculiarly, specific recovery performance rates were sometimes much better than global recovery performance rates of node harvest. This is reason for caution, as this indicates that node harvest's ensemble members can include relatively unimportant variables and hence produces incorrect prediction rules with split points for these variables. Depending on the amount of error allowed (Nagelkerke's R^2 value), the rule ensemble models (both RuleFit and PRE) sometimes had variable recovery performance rates similar to (in cells with less error) or slightly worse than random forests. Furthermore, global recovery performances of all methods improved when class outcomes were balanced ($P(Y = 1) = .5$).

Regarding the MRI dataset, the variable importances of all methods extracted similar features as the most important ones involved in distinguishing patients with Alzheimer's disease from controls. Moreover, a couple of these chosen predictors corresponded with findings in previous literature. This was a satisfactory result, given that it was not possible to assess global recovery performance on a real dataset and that the simulated global recovery performances were lower in settings with more noise and/or smaller training set sizes. Bearing in mind the simulation results, the most trustworthy model regarding variable importances in such a noisy and high-dimensional setting, is random forests.

Predictive performance of classifiers remained of great interest, especially in this study; as the generally high accuracy of random forests was taken as benchmark. Ideally, there should be a tree-based ensemble method with predictive performances similar to or better than random forests that can rely on interpretable, succinct prediction rules instead a black box prediction mechanism.

The benchmark random forests was indeed among the best predictors in the simulation study and was the best one in the MRI application. Yet depending on the simulation designs, some of the other methods of interests performed slightly better than random forests. In the simulation, OTE only contrasted with random forests on the error design factor regarding accuracy rates and Brier scores. With more signal OTE was slightly more accurate, but with more error random forests was more accurate. OTE also had significantly lower

AUC values than random forests in the simulation. The MRI data application was consistent with all of these findings; OTE came second to random forests on every model performance measure. Although node harvest has potential for being a more interpretable alternative than OTE, it was consistently worse than random forests on (almost) every measure, which was why this method contrasted with random forests on nearly every effect in the simulation. In the application study node harvest came third regarding model performances. Rule ensembles (both PRE and RuleFit) were more accurate than random forests in settings with more signal. Performances declined in noisier settings, which was especially evident in the application study. Although rule ensembles was an initially promising method, it had the worst performances in the MRI application. There seemed to be a trade-off between ensemble size or ensemble member complexity and predictive performance in high-dimensional settings; this is what node harvest and rule ensembles suffered of. Good discriminative abilities are paramount for e.g., identifying diseased individuals in medical applications. Interpretability with prediction rules help in explaining why a patient is at risk of being diseased, but unfortunately the studied methods cannot sufficiently help in providing such information if it is at the expense of predictive performances.

The changes in model performance and specific recovery performance depending on the design factors did not always follow the same patterns for node harvest and PRE. For example, predictive accuracy was higher in settings where class outcomes were unbalanced while global and specific recovery performances were lower in such settings (and vice versa). Earlier in the Results (Section 3.2.3) there was already a note stating that accuracy rates in those settings approximately correspond with $P(Y = 1)$: less classification mistakes can be made when one class strongly dominates. Recovery performance could have been worse in such settings as the presence of the dominating class 0 there is less support for rules predicting $Y = 1$. This mainly accounts for PRE, where interaction recovery was bad in the design cells with unbalanced class outcomes ($P(Y = 1) = .1$). Node harvest on the contrary had reasonable or good specific recovery performance in such settings and even had perfect conditional split point recovery in design cells where classes were unbalanced and there was no noise.

Nevertheless, in settings with full signal and balanced class proportions, PRE had the best variable interaction

recovery and predictive performances comparable to or better than random forests. This confirms that rule ensembles is a good alternative to random forests in settings with a lot of signal. It is a good combination of predictive performances comparable to the benchmark with the advantage of producing simple rules that contain correct information on the data structure. Rule ensembles is preferred over node harvest, as in beforementioned settings node harves had predictive performances that were much worse than those of rule ensembles (and thus random forests). An example of a suitable application of rule ensembles (with RuleFit) is described the study of Fokkema et al. (2015), where a rule ensembles model was fitted to clinical data on depression and anxiety. It resulted in a simple and interpretable model with predictive performances similar to traditional actuarial methods usually applied to such data (accuracy rate .653 and AUC .868 for RuleFit versus accuracy rate .659 and AUC .689 for logistic regression). There are however two points that should be taken into account. The first one is that there is uncertainty about whether conditional split point recovery in rule ensembles is better than node harvest, as we did not manage to assess split point recovery with PRE. The second (minor) point is that specific recovery performance could not be assessed with RuleFit and the model performance analyses showed that these two variations of rule ensembles did not behave entirely similarly. Hence there is no certainty about whether RuleFit would comply to the patterns in specific recovery performance results of PRE.

5.2) Novelties

In this study, we investigated four tree ensemble methods in detail, that make use of a type of binary partitioning algorithm like CART. Although the abilities of the four chosen methods have already been demonstrated in their original papers or in other papers in which these methods were applied, they have not yet been compared in the ways presented here.

In papers where machine learning methods are compared, usually only summaries of technicalities are given to explain the methods that are applied (e.g., Maroco et al. 2011). On the contrary, in this thesis the technicalities of our methods of interest were explained alongside a demonstration on a simple fake dataset to help the reader (or user) understanding the characteristics and modi operandi.

Newly proposed machine learning methods are commonly demonstrated through prediction assessment on simulated data and real datasets, where mainly the strengths in predictive performances are emphasized. This was also a great interest in this research, although only in combination with information on recovery performance. Other papers concentrate solely on e.g., variable selection properties of machine learning methods. For example, Strobl et al. (2007), and Strobl and Zeileis (2008) assessed variable importance properties and variable importances as tools for variable selection, but only for random forests. Here not only random forests variable importances, but also variable importances of three other tree ensemble methods were studied. We devised a way to compute variable importances for node harvest, which was previously unavailable. This was based on variable importance measures of rule ensembles, using nodes as analogues of rules. While usually only one or two interests are assessed per paper, in this thesis predictive performances, variable importance assessments, and, where possible, variable threshold recovery were assessed all at once. Everything was compared with each other to find a classifier with a good balance between prediction and recovery performance.

Besides the toy data demonstrations and performance assessments through a simulation and application, this paper researched interpretational potentials of tree ensemble methods differently than the orignial papers of these methods. For example, Meinshausen (2010) highlighted the interpretational characteristics of node harvest as tree-based ensemble through an example with a prediction done on an individual observation. The interpretational potential of rule ensembles has been explained in Friedman and Popescu (2008) through a technical explanation of how rules are decomposed trees and how parameters or settings can be specified to control ensemble member complexity. These papers explained how individual ensemble members contribute to a certain outcome (i.e., prediction rules). However, they did not assess the correctness of the information contained in the individual ensemble members. In our study interpretation was investigated as not just the simplicity (or complexity) of individual ensemble members. The ability of these members to retrieve the correct structure of underlying data is just as important as elaborating on characteristics such as simplicity of ensemble members or prediction rules. The toy data demonstration and simulation with (specific) recovery performance assessment were a first to assess the ability of tree-based ensemble rules to correctly recover the true data structure. Furthermore, we especially focused on the information contained in the most important

ensemble members, which was not explicitly done in e.g., Meinshausen (2010), or Friedman and Popescu (2008).

Lastly, OTE, node harvest, and rule ensembles have not yet been assessed on a high-dimensional classification problem. Khan et al. (2016) suggested that OTE is especially useful for the application to high-dimensional datasets in combination with a feature selection method (such as proportional overlapping score (POS); Mahmoud et al. 2014), although they did not actually demonstrate this in their paper. Node harvest has been demonstrated on a high-dimensional dataset for regression (Meinshausen 2010), but random forests outperformed it there. To our knowledge, the application of these methods to a Alzheimer's disease dataset in particular, as used here, was a first.

5.3) Suggestions for improvement

Certain suggestions can be done for expansion of the simulation set-up. First of all, training set sizes were relatively or very large. Smaller training set sizes should be included as well, as this is representative for dataset sizes in certain fields of the life or behavioral sciences, where the number of available subjects is limited. Secondly, the noise factor can include a much noisier setting. Both the simulation and application results indicated that random forests performed better than the other methods in the settings with more error and the interpretation-focused methods perform worse regarding predictive and information recovery performance. Thirdly, an optional extra factor, the number of predictors M, can be implemented in the simulation as well. In this research, $M = 10$ was retained for the simulation. A combination of a large value for M with a small value for n_{train} makes a high-dimensional design setting. This with the addition of a noisier design factor, would aid assessment of the trustworthiness of the final models that were fitted on the Alzheimer's disease dataset. Obviously, the simulation should also include conditional split point assessments of rule ensembles (PRE).

Another suggestion is to expand the research by implementing regression besides (binary) classification to compare patterns in performance changes. The original papers describing OTE, node harvest, and rule

ensembles mainly demonstrated these methods on datasets for regression. In Meinshausen (2010) for example, node harvest performed better than random forests (in terms of explained variance) when applied to regression data with additional noise. This contradicts our findings for noisy settings in binary classification. The lower performance of node harvest in the current binary classification settings could be due to the fact that nodes are not selected on purity, but rather on averages (here: proportions), weights are chosen by minimization of a quadratic loss, and hence prediction is based on weighted averages, rather than e.g., aggregation through votes such as in random forests and OTE.

A final note should be made about the analyses of the repeated cross-validated model performances, which was done with paired nonparametric testing. The repeats were treated as independent observations - an assumption for a Wilcoxon signed rank test - even though the observations were correlated because every cross-validation repetition was done on the same dataset. This could have affected the estimates of the ranks of the performances per method, leading to potentially underestimated confidence intervals for sensitivity and specificity. In future assessments of machine learning methods on this dataset, a different (repeated) cross-validation procedure should be considered for fairer comparison of method performances.

5.4) Conclusion

We aimed to find a tree-ensemble method with interpretational ease of a CART tree instead of providing a black box on variable involvement in predictions like random forests. This study shows that tree-based ensemble methods with more interpretability (simple ensemble members and smaller total ensemble sizes) can have predictive performances comparable with the generally well-performing method random forests, although only in certain settings.

Random forests as benchmark method was still the best classifier and had the best variable importance recovery. OTE showed overall similar performances compared with random forests. However, exactly because it is a reduced random forest with full trees, this method can also be regarded as a black box predictor that keeps us in the dark. Node harvest and rule ensembles are very suitable for interpretation, as they produce

relatively small ensembles with simple members, which are rules that provide insight in how certain outcomes are predicted. Node harvest generally was the worst predictor, but proved to be good in identifying important variable thresholds. In datasets with a lot of signal (i.e., Nagelkerke's $R^2 > .5$), rule-based ensemble methods are the clear winner: they have global recovery and predictive performances similar or better than random forests. Most importantly, their fitted models produce simple and interpretable rules containing more correct information about the data structure, as measured by the (specific) recovery performance criteria.

Despite these promising results of the rule-based models, the rule-based ensemble methods only partially solve the main problem as they only work excellently in datasets with a lot of signal. Especially high-dimensional or small sample-sized datasets still benefit from more complex and numerous ensemble members, which is characteristic for the benchmark random forests. Hence, among the studied methods, there is no perfect, more interpretable tree-based method with predictive performance competitive to random forests that can be used in these more complex settings, which makes the produced output for interpretation less reliable. We are still lost in the forest during the ongoing search for a 'perfect' tree-based ensemble learner that shows a good balance between prediction properties and interpretation possibilities.

References

Bernard, Simon, Laurent Heutte, and Sebastien Adam. 2009a. "On the Selection of Decision Trees in Random Forests." In *2009 International Joint Conference on Neural Networks*. Institute of Electrical; Electronics Engineers (IEEE). doi:10.1109/ijcnn.2009.5178693.

Bernard, Simon, Laurent Heutte, and Sébastien Adam. 2009b. "Influence of Hyperparameters on Random Forest Accuracy." In *International Workshop on Multiple Classifier Systems*, 171–80. Springer.

Bradley, Andrew P. 1997. "The Use of the Area Under the Roc Curve in the Evaluation of Machine Learning Algorithms." *Pattern Recognition* 30 (7). Elsevier: 1145–59.

Breiman, Leo. 1996. "Bagging Predictors." *Machine Learning* 24 (2). Springer Nature: 123–40. doi:10.1007/bf00058655.

———. 2001. *Machine Learning* 45 (1). Springer Nature: 5–32. doi:10.1023/a:1010933404324.

Breiman, Leo, Jerome Friedman, Richard Olshen, and Charles Stone. 1984. *Classification and Regression Trees*. Monterey, CA: Wadsworth; Brooks.

Brier, Glenn W. 1950. "Verification of forecasts expressed in terms of probability." *Monthly Weather Review* 78 (1). American Meteorological Society: 1–3. doi:10.1175/1520-0493(1950)078<0001:vofeit>2.0.co;2.

Cohen, Jacob. 1988. *Statistical Power Analysis for the Behavioral Sciences*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Faraway, Julian J. 2006. *Extending Linear Models with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models*. Chapman & Hall/CRC.

Fawcett, Tom. 2006. "An Introduction to ROC Analysis." *Pattern Recognition Letters* 27 (8). Elsevier BV:

861–74. doi:10.1016/j.patrec.2005.10.010.

Fokkema, Marjolein. 2016. *Pre: Prediction Rule Ensembles*. https://CRAN.R-project.org/package=pre.

Fokkema, Marjolein, Niels Smits, Henk Kelderman, and Brenda W. J. H. Penninx. 2015. "Connecting Clinical and Actuarial Prediction with Rule-Based Methods." *Psychological Assessment* 27 (2). American Psychological Association (APA): 636–44. doi:10.1037/pas0000072.

Friedman, Jerome H., and Bogdan E. Popescu. 2003. "Importance Sampled Learning Ensembles." Stanford University, Department of Statistics.

———. 2008. "Predictive Learning via Rule Ensembles." *The Annals of Applied Statistics* 2 (3). Institute of Mathematical Statistics: 916–54. doi:10.1214/07-aoas148.

———. 2012. *RuleFit with R* (version 3). http://statweb.stanford.edu/~jhf/R_RuleFit.html.

Gerds, Thomas A., Tianxi Cai, and Martin Schumacher. 2008. "The Performance of Risk Prediction Models." *Biometrical Journal* 50 (4). Wiley-Blackwell: 457–79. doi:10.1002/bimj.200810443.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning*. Springer New York. doi:10.1007/978-0-387-84858-7.

Head, D., Randy L. Buckner, Joshua S. Shimony, Laura E. Williams, Erbil Akbudak, Thomas E. Conturo, Mark McAvoy, John C. Morris, and Abraham Z. Snyder. 2004. "Differential Vulnerability of Anterior White Matter in Nondemented Aging with Minimal Acceleration in Dementia of the Alzheimer Type: Evidence from Diffusion Tensor Imaging." *Cerebral Cortex* 14 (4). Oxford University Press (OUP): 410–23. doi:10.1093/cercor/bhh003.

Hothorn, Torsten, Kurt Hornik, and Achim Zeileis. 2006. "Unbiased Recursive Partitioning: A Conditional Inference Framework." *Journal of Computational and Graphical Statistics* 15 (3). Informa UK Limited:

651–74. doi:10.1198/106186006x133933.

Kester, M. I, and P. Scheltens. 2009. “Dementia: THE BARE ESSENTIALS.” *Practical Neurology* 9 (4). BMJ: 241–51. doi:10.1136/jnnp.2009.182477.

Khan, Zardad, Asma Gul, Osama Mahmoud, Miftahuddin Miftahuddin, Aris Perperoglou, Werner Adler, and Berthold Lausen. 2016. “An Ensemble of Optimal Trees for Class Membership Probability Estimation.” In *Analysis of Large and Complex Data*, 395–409. Springer Nature. doi:10.1007/978-3-319-25226-1_34.

Khan, Zardad, Asma Gul, Aris Perperoglou, Osama Mahmoud, Werner Adler, Miftahuddin, and Berthold Lausen. 2015. *OTE: Optimal Trees Ensembles for Regression, Classification and Class Membership Probability Estimation*. https://CRAN.R-project.org/package=OTE.

Kuhn, Max, Jed Wing, Stew Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, et al. 2016. “Caret: Classification and Regression Training. R Package Version 6.0-68.” https://CRAN.R-project.org/package=caret.

Latinne, Patrice, Olivier Debeir, and Christine Decaestecker. 2001. “Limiting the Number of Trees in Random Forests.” In *Multiple Classifier Systems*, 178–87. Springer Nature. doi:10.1007/3-540-48219-9_18.

Liaw, Andy, and Matthew Wiener. 2002. “Classification and Regression by RandomForest.” *R News* 2 (3): 18–22. http://CRAN.R-project.org/doc/Rnews/.

Mahmoud, Osama, Andrew Harrison, Aris Perperoglou, Asma Gul, Zardad Khan, Metodi V Metodiev, and Berthold Lausen. 2014. “A Feature Selection Method for Classification Within Functional Genomics Experiments Based on the Proportional Overlapping Score.” *BMC Bioinformatics* 15 (1). Springer Nature: 274. doi:10.1186/1471-2105-15-274.

Maroco, João, Dina Silva, Ana Rodrigues, Manuela Guerreiro Isabel Santana, and Alexandre de Mendonça. 2011. “Data Mining Methods in the Prediction of Dementia: A Real-Data Comparison of the Accuracy, Sensitivity and Specificity of Linear Discriminant Analysis, Logistic Regression, Neural Networks, Support

Vector Machines, Classification Trees and Random Forests." *BMC Research Notes* 4 (99). doi:10.1186/1756-0500-4-299.

Meinshausen, Nicolai. 2010. "Node Harvest." *The Annals of Applied Statistics* 4 (4). Institute of Mathematical Statistics: 2049–72. doi:10.1214/10-aoas367.

———. 2015. *NodeHarvest: Node Harvest for Regression and Classification.* https://CRAN.R-project.org/package=nodeHarvest.

Möller, Christiane, Anne Hafkemeijer, Yolande A.L. Pijnenburg, Serge A.R.B. Rombouts, Jeroen van der Grond, Elise Dopper, John van Swieten, et al. 2015. "Joint Assessment of White Matter Integrity, Cortical and Subcortical Atrophy to Distinguish AD from Behavioral Variant FTD: A Two-Center Study." *NeuroImage: Clinical* 9. Elsevier BV: 418–29. doi:10.1016/j.nicl.2015.08.022.

Nagelkerke, Nico JD. 1991. "A Note on a General Definition of the Coefficient of Determination." *Biometrika* 78 (3). Oxford University Press: 691–92.

R Core Team. 2016. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Richardson, John T.E. 2011. "Eta Squared and Partial Eta Squared as Measures of Effect Size in Educational Research." *Educational Research Review* 6 (2). Elsevier BV: 135–47. doi:10.1016/j.edurev.2010.12.001.

Sing, T., O. Sander, N. Beerenwinkel, and T. Lengauer. 2005. "ROCR: Visualizing Classifier Performance in R." *Bioinformatics* 21 (20): 7881. http://rocr.bioinf.mpi-sb.mpg.de.

SPSS, IBM, and others. 2015. "IBM Spss Statistics for Windows, Version 23.0." *New York: IBM Corp.*

Strobl, Carolin, and Achim Zeileis. 2008. "Danger: High Power!–exploring the Statistical Properties of a Test for Random Forest Variable Importance."

Strobl, Carolin, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. 2007. "Bias in Random

Forest Variable Importance Measures: Illustrations, Sources and a Solution." *BMC Bioinformatics* 8 (25). doi:10.1186/1471-2105-8-25.

Sullivan, Gail M., and Richard Feinn. 2012. "Using Effect Sizeor Why the P Value Is Not Enough." *Journal of Graduate Medical Education* 4 (3). Journal of Graduate Medical Education: 279–82. doi:10.4300/jgme-d-12-00156.1.

Versace, Amelia, Jorge R. C. Almeida, Stefanie Hassel, Nicholas D. Walsh, Massimiliano Novelli, Crystal R. Klein, David J. Kupfer, and Mary L. Phillips. 2008. "Elevated Left and Reduced Right Orbitomedial Prefrontal Fractional Anisotropy in Adults with Bipolar Disorder Revealed by Tract-Based Spatial Statistics." *Archives of General Psychiatry* 65 (9). American Medical Association (AMA): 1041. doi:10.1001/archpsyc.65.9.1041.

Wright, Marvin N., Andreas Ziegler, and Inke R. König. 2016. "Do Little Interactions Get Lost in Dark Random Forests?" *BMC Bioinformatics* 17 (1). Springer Nature. doi:10.1186/s12859-016-0995-8.