Article

Carolin Strobl*, Yannick Rothacher, Sven Theiler and Mirka Henninger

Detecting interactions with random forests: a comment on Gries' words of caution and suggestions for improvement

https://doi.org/10.1515/cllt-2024-0028
Received March 11, 2024; accepted October 11, 2024; published online November 11, 2024

Abstract: Tree-based methods are being both successfully applied and critically discussed in corpus linguistics. In this article, we would like to contribute a few aspects to this discussion from a methodological point of view. These aspects include the interpretation of interaction effects in single trees and random forests, as well as more general aspects like stability and overfitting. In particular, we have conducted a simulation study to investigate an approach suggested by Gries for computing the importance of interactions in random forests more systematically than the previous literature. The evidence of this simulation study shows that, even when interaction predictors are explicitly added, the permutation variable importance is not suited for distinguishing between main effects and interaction effects or between interaction effects of different orders. We also discuss the use of partial dependence (PD) and individual conditional expectation (ICE) plots for illustrating the functional form and potential interaction effects, and other means of interpretable machine learning.

Keywords: classification and regression trees; ensemble methods; interaction effects; variable importance; random forests; interpretable machine learning

***Corresponding author: Carolin Strobl**, Department of Psychology, University of Zürich, Zürich, Switzerland, E-mail: carolin.strobl@uzh.ch
Yannick Rothacher, Department of Psychology, University of Zürich, Zürich, Switzerland
Sven Theiler, School of Applied Psychology, University of Applied Sciences and Arts Northwestern Switzerland, Olten, Switzerland
Mirka Henninger, Faculty of Psychology, University of Basel, Basel, Switzerland

1 Introduction

In recent years, corpus linguistics has seen an increased use of advanced statistical analysis methods. While (generalized) linear models still remain the most commonly applied approach, researchers have become increasingly interested in other approaches, including methods from the field of machine learning. Especially the family of tree-based methods has emerged as an alternative to classical parametric statistics.

Early implementations of classification and regression trees include CART (Breiman et al. 1984) and C4.5 (Quinlan 1986, 1993). To overcome some of the limitations of individual trees, the ensemble methods bagging (Breiman 1996) and random forests (Breiman 2001) have been introduced and implemented by Leo Breiman and Adele Cutler around the turn of the millennium.

These traditional tree and forests algorithms are still widely used, despite the fact that they show a preference for predictor variables offering many possible cutpoints – regardless of their information content (e.g., Kim and Loh 2001; Strobl et al. 2007a; White and Liu 1994). This unwanted behavior is termed variable selection bias and also carries forward to variable importance measures for bagging and random forests (Strobl et al. 2007b). Modern implementations of trees and forests by Loh and Shih (1997) and Hothorn et al. (2006) have overcome this issue. In the R system for statistical computing, two implementations of conditional inference trees (Hothorn et al. 2006) and random forests based on conditional inference trees[1] are available in the `party` and `partykit` packages and provide unbiased variable selection when used with the default settings (Hothorn and Zeileis 2015; Hothorn et al. 2006, 2024a, 2024b; Strobl et al. 2007b).

These modern implementations of trees and forests have been made known, and also critically assessed, in corpus linguistics by, e.g., Tagliamonte and Baayen (2012), Szmrecsanyi et al. (2016), Gries (2020), Gries (2021), and Bernaisch (2022). The critical discussion of tree based methods in corpus linguistics is much more sophisticated than in many other application areas of these methods, such as medicine or genetics. We highly appreciate this discussion culture, and would like to contribute to this discussion from a methodological point of view. In particular, we will take up and comment on a few remarks and a methodological suggestion in Gries (2020) and Gries (2021), and also provide references to related methodological literature for the interested reader. Note, however, that a complete introduction to trees and

1 Please note that in the remainder of this text we will use the term random forests as an umbrella term of all types of random forests, including random forests based on conditional inference trees, unless explicitly stated otherwise.

forests is beyond the scope of this text, so that readers who are not yet familiar with the topic should first consult introductory texts, such as Strobl et al. (2009b) and Gries (2021).

2 Properties and pitfalls of trees and forests

Traditional tree algorithms are based on descriptive criteria, such as the Gini index, for selecting the optimal splitting variable and cutpoint in one step. This leads to the undesired variable selection bias pointed out above. Modern tree algorithms have disentangled variable and cutpoint selection, and base the variable selection step on statistical significance tests. This procedure not only solves the problem of variable selection bias. It also means that, while traditional trees required pruning (as is very well explained in Gries 2021), modern trees can use the same statistical significance tests as stopping criteria to regulate tree depth. However, users should be aware that – just like any other statistical significance tests – the tests employed in modern tree algorithms have a higher statistical power for detecting effects in larger samples (cf. Henninger et al. 2023b, for an example from psychology).

Due to the fact that trees only ever consider one splitting variable at a time,[2] apparently simple patterns in the data generating process can be very hard to capture for a tree. In particular, a pattern where two variables have a perfect interaction effect but no main effects (termed an XOR problem in the machine learning literature) is hard to impossible to detect for a tree. Another thing that trees are not good at is approximating linear functions, as is pointed out by Gries (2020). This can be seen as a downside of the exploratory nature of trees. They are not provided with as much structure as, for example, linear models. On the other hand, this also means that they are not forced to stick to a provided structure – which may be too restrictive to describe the true pattern in the data.

Trees are able to approximate any functional form given enough data. The result, a piecewise constant function, may not look very elegant (ensemble methods like random forests do a much better job of approximating functional forms more smoothly), but the advantage of such an exploratory approach in general is that it can detect patterns in the data that were not known to or hypothesized by the

2 For the first split in each tree, this corresponds to assessing the strength of the marginal main effect of each potential predictor variable. For any subsequent split, this corresponds to assessing the strength of each potential predictor conditional on and in interaction with any previous splits, but regardless of any splits yet to come. In this sense, the tree building process is only locally optimal, and does not necessarily lead to the globally optimal model, as is also mentioned by Gries (2020).

researchers. In contrast, a linear model will fit only a linear function unless otherwise specified and will miss, for example, a quadratic effect in the data if a quadratic term is not explicitly added to the model.

2.1 From trees to ensemble methods

A problematic property of trees, that is often mentioned in the literature and has led to the development of ensemble methods like random forests, is their instability to small changes in the data. While Gries (2020) mentions that these changes can affect both the prediction and the tree structure, Philipp et al. (2016, 2018) point out that quite different-looking tree structures can actually lead to essentially the same predictions. This highlights that often the tree structure is over- or misinterpreted. For example, while the variable used for the first split in a tree is the one that showed the strongest main effect, it is the entire pattern of the predictions in the end nodes that shows whether and how several splits in the same variable approximate the functional form (as illustrated for a linear effect approximated by several splits in a tree by Gries 2020, Fig. 3) or whether the first variable works together with the variables below to form an interaction effect (as illustrated by Strobl et al. 2009b, Fig. 4).

Ensemble methods like random forests (Breiman 2001) and their predecessor method bagging (Breiman 1996) have overcome the instability issue by means of averaging predictions over several hundreds or thousands of trees. This makes the predictions of ensembles much more smooth and typically more accurate than those of single trees, but comes at the price that one looses the interpretability that is inherent to individual trees (as long as they are not too large, as Gries 2020, points out). This lack of interpretability refers to the fact that random forests do not offer direct insights into the relationship between the individual predictor variables and the response variable. It remains unclear whether and how the individual predictor variables contribute to the prediction of the response variable alone and/or in interactions. For this reason, in the machine learning literature ensemble methods are often termed black box methods (a metaphor for the fact that one enters the predictor variables and out comes the prediction – but what happened in between is obscure).

2.2 Overfitting and related issues

The way that ensembles of bagged trees and random forests are constructed – by fitting individual trees on bootstrap samples with replacement or (preferably for unbiased variable selection, as shown by Strobl et al. 2007b) subsamples without

replacement from the original data – also has the advantage that each tree comes with its own, built-in test data set, namely those observations that were not used for training the respective tree (termed out-of-bag, or OOB, data). This is a very useful property of random forests, which Gries (2020) rightly draws attention to. However, since the terminology used by Gries (2020) is not entirely in line with the statistical and machine learning literature, we would like to elaborate on this topic.

To explain why it is important to assess the quality of machine learning methods on data that were not used for fitting the model, let's first review how models are compared and assessed in classical, parametric statistics. For example, imagine that you want to use a linear regression model to predict a response variable from several potential predictor variables, but do not know in advance which predictors are and are not informative for predicting the response (or whether they work in linear main effects, nonlinear or interaction effects). Imagine you estimated two regression models for predicting the response variable: model 1, containing only predictors A and B, and model 2, containing predictors A, B, C, D, E and F. How can you decide which model is better? While, for example, the simple R^2 statistic for linear regression tends to increase with the number of added variables, even if spurious effects are added, the adjusted R^2 (but also F-tests or likelihood-ratio-tests for model comparisons in nested models) will account for the model complexity.[3] So in the world of parametric statistics, there are clear-cut criteria for deciding whether model 2 is better than model 1, which balance the explained variance against the model complexity. But these clear-cut criteria are based on certain mathematical assumptions.

Machine learning methods come with far fewer assumptions than classical parametrical statistical approaches. This also means that we no longer have the entire toolbox of statistical significance testing and confidence intervals available for assessing machine learning solutions. Without this toolbox, the only information we have about how good one machine learning model performed compared to another one is their prediction accuracy (which for categorical response variables is also termed classification accuracy).

Now, similarly to what would happen if you would compare the un-adjusted R^2 of two regression models, if you compared the prediction accuracies of two machine learning models (again model 1 containing only predictors A and B, model 2 containing A, B, C, D, E and F) on the original data, the more complex model will typically perform better, just because it is more flexible. You can imagine this when you think about predicting the scores of your students in an exam based on a few truly relevant

3 F- and likelihood-ratio-tests will even provide p-values, indicating whether the improvement of one model over the other is significant, i.e., higher than expected merely due to random sampling fluctuations.

predictors (how long did they study for the exam – probably helpful for getting a high score; did they attend a party the night before the exam – probably not so much). But your data set may also contain other potential predictor variables (do your students have a cat or dog; what is their shoe size; what is the shoe size of their grandmother; … – you get our point, these variables are not informative for the students' exam performance). If you would use a regression or machine learning model containing only the two truly influential predictors (study time and party attendance) the model would show a decent prediction accuracy, but not a perfect one. Remaining unexplained differences between the students' exam scores could be due to other, unobserved characteristics, such as their IQ or previous experience with the course topic, but also how they feel that particular day, if they made any careless errors etc. Still, this model would be very useful, because it does not only decently predict the exam scores of this year's students, but it will also decently predict the grades of next year's students, as long as the general mechanisms underlying your course and exam don't change.

If, however, in addition to the truly informative variables you included several irrelevant variables in your model (such as pet ownership and shoe sizes), what would happen is the following: The model would show a higher prediction accuracy for this year's students. The reason for this is that by means of the additional irrelevant variables, you can make more fine-grained predictions. In the extreme case you would include so many irrelevant variables that only one particular student is described by each combination. For example there might be only one student who studied dozens of hours, did not attend a party the night before the exam, has two cats, shoe size 38, a grandmother with very large feet, etc. A model that is extremely flexible because it contains so many predictors will learn what exam score this particular student achieved, and predict exactly that same exam score for all students with the same values on these predictors. This is actually a very good strategy for predicting the exam score of this particular student in the data set from this particular year, and the model will achieve a very high prediction accuracy on this year's data this way. However, we would not expect this prediction to work very well for next year's students – or generally speaking, another sample from the same population. When a model is too flexible and thus adheres too strongly to random variations in the original sample it does not generalize well to other samples from the same population. This is termed overfitting.

Due to the stabilizing effect of averaging over the individual trees, random forests tend to be less affected by overfitting than other machine learning methods. In Gries (2020) this point is correctly made. However, presumably for didactic reasons, the terminology about predictions used by Gries (2020) is not the one common in the statistics and machine learning literature. To avoid confusion, we would like to point out that: (i) In the statistics and machine learning literature, the term

prediction describes the process of entering the predictor variable values of a real or hypothetical observation into a fitted model and receiving the model's prediction of the response variable for that observation. This can be done for individual or all observations in a data set, and for the same data set that the model was fitted on (termed *training data* or *learning data*) or new data (termed *test data*). (ii) The *prediction accuracy* is computed by comparing the predicted and true values of the response variable and aggregating over all observations. When the response variable is categorical, the prediction accuracy is also termed *classification accuracy*. It is typically described by the percentage of observations for which the predicted class is equal to the true class.[4]

In contrast to these conventions, Gries (2020) calls the accuracy of out-of-bag predictions of a random forest (which are based on different observations than those the forest was fitted to, similar to a fresh test sample) "prediction accuracy", while he calls the prediction accuracy of a tree on the original learning data "classification accuracy". We fully agree that the prediction accuracy of out-of-bag predictions of a random forest gives a better estimate of the prediction accuracy that is to be expected for other samples from the same population. At the same time, in order to avoid confusion for readers consulting articles or textbooks from statistics or machine learning, we would like to point out that the contrasting use of the terms "prediction" versus "classification" accuracy in Gries (2020) is not in line with how these terms are used in the statistics and machine learning literature.

2.3 Interpretability and stability

Coming back to the interpretability (or lack thereof) of random forests, we agree with Gries (2021) that when the true pattern in the data is more complex, trying to interpret a random forest by means of fitting an additional single tree to the data will produce an oversimplification. Below, we will discuss more adequate means of interpretable machine learning. However, we would also like to point out that when the true pattern in the data is simple, a single tree may be all that it takes to capture this pattern – and in this case, the interpretability of a single tree is a big advantage over the black box property of a random forest. Whether a specific data set may contain a pattern that is sufficiently described by a single tree can be explored by means of the stability diagnostics available in the `stablelearner` package in R

4 Some classification algorithms are also able to return predicted class probabilities, rather than predicted class memberships, but these can also be compared to the true class, for example by means of the Brier score or by the area under a receiver operating characteristic (or ROC) curve. For metric response variables, the prediction accuracy can be measured, e.g., by the mean squared distance between predicted and true response values (termed mean squared error or MSE).

(Philipp et al. 2016, 2018, 2023). In this framework, the variable and cutpoint selection of the tree fit on the original data is compared to that of trees fit on hundreds of samples from the same data (similar to a random forest, but here the original tree has a special role). If the same variables and similar cutpoint locations as in the original tree are selected by the majority of resampling-based trees, the results can be considered as stable and it is safe to interpret the original tree. If not, the complexity of the pattern in the data will be better captured by an ensemble method, and in that case no single tree is suitable for describing the pattern.

2.4 Tuning and runtime

Random forests are often said to "work well off the shelf", i.e., with the default settings for the tuning parameters. Still, the user should always double check the default settings of each random forest implementation they are using. In particular, while in the `randomForest` (Liaw and Wiener 2022) function the default value of the `mtry` argument is set to the default values suggested by Breiman for the original implementation (for classification to the square root of the number of predictor variables, for regression to one third of the number of predictor variables), in `party` the `mtry` argument is an arbitrary fixed number. As correctly highlighted by Gries (2021) for `randomForest`, it is often worthwhile to tune (i.e., to optimally select) the value of `mtry`. The same holds for `cforest`. For `cforest` from the `party` package, tuning based, e.g., on cross validation is available in the `caret` package (Kuhn 2008; Kuhn et al. 2023).

Cross validation means that the sample is randomly split into k parts, of which $k-1$ parts are used for fitting the model and the remaining part is held back to assess the prediction accuracy on fresh data. This process is repeated k times until all observations have been used in turn for fitting the model and for assessing the prediction accuracy. The entire procedure is conducted for several different values of the to-be-tuned parameter, and the value with the best cross-validated prediction accuracy is chosen. The number of trees in the forest, `ntree`, is typically not tuned this way, because, while for `mtry` values smaller or larger than the optimal value can lead to suboptimal performance, increasing the number of trees will only make the results more accurate and more stable – only the extent of the increase in accuracy flattens after a certain point. However, when the number of trees is chosen too small, one might not yet have reached the point where the results are sufficiently stable to, e.g., interpret random forest variable importance scores. To check whether the number of trees is sufficiently large, Strobl et al. (2009a) recommend to re-fit a random forest and re-compute the variable importances using a different seed

(which determines the status of the random number generator in R). If the results vary notably, the number of trees was not yet sufficiently high.

Since for larger data sets runtime can be an issue, it may be worth mentioning that in the `partykit` package the functions for fitting conditional inference trees and forests were written entirely in R (while in `party` parts of their implementation were outsourced to C). This makes `partykit` very flexible, but means that, at the moment, fitting conditional inference trees and forests in the older package `party` is actually faster than in the newer package `partykit`. For computing the conditional permutation importance (see also below), the `permimp` package (Debeer et al. 2021) provides a faster implementation.

2.5 Variable importance and functional form

Random forest variable importance measures can give a first impression which variables were relevant for the prediction. They have been motivated rather heuristically and their absolute values depend on various factors beyond the true effect of a variable, so that they are not directly comparable between different data sets. Variable importance scores should therefore merely be interpreted in a qualitative fashion. For example, when a few variables show much higher importance scores than the rest, it makes sense to concentrate on these variables in future studies.[5] However, researchers often desire clear-cut decisions which variables are "significantly important". Rothacher and Strobl (2023) review different heuristics for significance tests for random forest variable importance measures. In particular, they show that the rule of thumb suggested by Strobl et al. (2009b), that has been erroneously communicated by some researchers as if it was a significance test, does not possess the properties of a formal statistical test. This rule of thumb is based on the idea that – in a sparse setting, where the number of potential predictors is large, but only few are truly informative – the importance scores of the non-informative predictors will randomly fluctuate around zero. In this kind of setting, which is common, e.g., in genetics, but probably not so common in linguistics, the absolute value of the strongest negative importance can serve as a rule-of-thumb lower bound, in the sense that variables not even exceeding this bound can be considered as noise variables. It is, however, not a formal significance test, and was not presented as one by Strobl et al. (2009b). When nevertheless used as such, Rothacher and Strobl (2023) show that it has an inflated type I error rate. Interestingly, some

5 As always when using data to inform variable selection or model selection in general, the data used for this exploratory step must not be re-used for any confirmatory modelling. This would lead to false positive results.

other, computationally much more intensive approaches show even worse statistical properties.

In addition to the original permutation variable importance, a conditional permutation variable importance is available (Debeer and Strobl 2020; Strobl et al. 2008). As is very well explained by Gries (2021), the conditional permutation scheme takes correlations between predictor variables into account and thus avoids "exaggerating the importance of predictors that are correlated with other [informative] predictors" (Gries 2021, p. 470, with our addition in square brackets). However, the way the conditioning is implemented at the moment leads to increased computation times and also overcorrects in the sense that the conditional permutation importance of all correlated predictors is lowered compared to that of uncorrelated predictors (Henninger et al. 2023a). While work is being done currently to further investigate and improve the behavior of the conditional permutation importance, we would like to point out that neither the unconditional nor the conditional permutation importance should be considered as "the truth", but rather as different points on the marginal-to-partial continuum (Debeer and Strobl 2020) and that the most information lies in their comparison rather than either result alone.

Variable importance scores indicate descriptively, which variables – alone or in interactions – contributed to the prediction of the random forest. For a more thorough interpretation it would be desirable to learn more about which variables contribute in main effects versus interactions and also in which functional form. Here we would like to clarify two misconceptions, but first need to review a few important terms and concepts. When you think of a linear or logistic regression model, everything is very tidy, in the sense that there are individual coefficients quantifying the linear, quadratic, or other curvilinear effect of a single variable or the interaction effect of two or more variables. As was mentioned above, in a single tree, the tree structure together with the predictions in the end nodes describe these effects and the approximation of the functional form captured by the tree – at least to the trained eye. In a random forest with hundreds or thousands of trees it is not possible to visually captures these effects. There are, however, additional approaches from the field of interpretable machine learning, that provide additional hints. Partial dependence plots (Apley and Zhu 2020; Henninger et al. 2023a, for an introduction) or other plots of the predicted responses for different values of the predictors, such as those suggested by Szmrecsanyi et al. (2016), Gries (2020) and Hundt et al. (2020), are one possibility to illustrate the functional form of the association between a single predictor variable and the prediction for the response variable, typically averaging over the values of all other predictors. It is important to be aware that the information about the functional form of the association between predictor and response, that is approximated rather smoothly by a random forest and then

graphically displayed in a partial dependence plot, is a valuable information, because in exploratory modelling the functional form of the association between predictors and response is learned from the data rather than pre-specified by the user.

Still, a univariate partial dependence plot is an approximation of the functional form of the main effect of only one predictor variable at a time. So while Gries (2021) argues correctly that partial dependence plots contain more information than, say, bivariate χ^2-tests between each individual predictor and the response – namely the information on the functional form – his statement on page 467 ("…it also makes no sense to summarize a multifactorial forest with monofactorial tables – this is what we use partial dependence plots for.") should not be misinterpreted in the sense that univariate partial dependence plots could capture interaction effects.

We would like to illustrate this with a simple example: Figure 1 shows partial dependence plots for two predictors from a random forest. We see that the predicted response is essentially the same over the range of the hypothetical numeric (left) and the two values of the binary (right) predictor variable. If there was a strong main effect, the predicted response values should show a visible monotone or non-monotone pattern, but these two predictors seem to be irrelevant – when looking at one of them at a time, marginalizing over all other predictors. We will come back to this example soon, but first would like to provide a little more detail about how partial dependence plots are constructed.

Partial dependence plots display the average predicted value of the response variable (on the *y*-axis) as a function of the value of a predictor variable (on the *x*-axis). In order to create a partial dependence plot, the average predicted value of the response variable needs to be computed for different values of the predictor

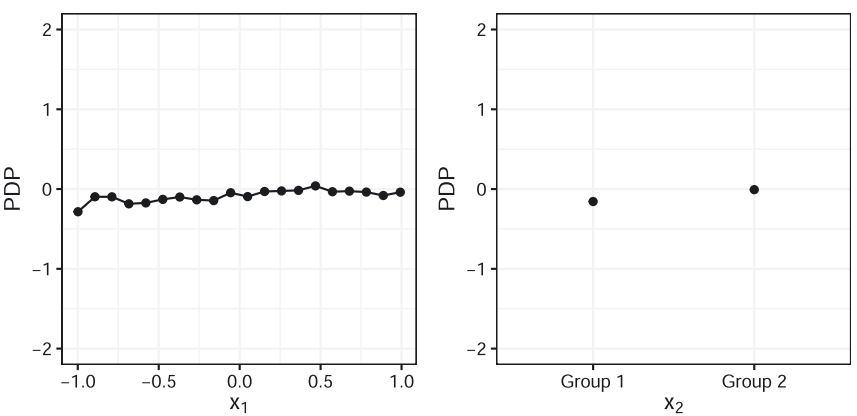

Figure 1: Partial dependence plots for two predictor variables, x_1 (left) and x_2 (right).

variable.[6] In order to compute the average predicted value, first a prediction for each single observation in the dataset needs to made by the fitted random forest (or another machine learning model) for each of the different values of the predictor variable. The predictions for the single observations are then averaged over all observations in a second step.

One observation here corresponds to one row in the dataset. In many machine learning applications, every row in the dataset represents one person, and every column represents one variable, such as age, gender, economic status or wellbeing. Imagine that a random forest is used to predict the value of the response variable wellbeing from the values of the predictor variables age, gender and economic status. The partial dependence plot then has the following rationale: For making the prediction for a single observation, the observed values of all other predictor variables for this observation are used – only for the variable of interest the value is varied. For example, in order to create a partial dependence plot for the predictor variable age, the prediction for a single person will use that person's observed values for gender and economic status, but try out different values for the person's age. The random forest's predicted value of the response variable, wellbeing, is then computed for each value of age. These predictions show the levels of wellbeing predicted by the model for a certain person if we could hold all their other properties constant but vary their age to make them younger or older.

For every value of age, the predictions are then averaged over all persons in the second step to create the average prediction. In this sense, partial dependence plots average or marginalize over all other predictor variables. The average prediction (on the *y*-axis) is displayed in the partial dependence plot as a function of the different values of age that were tried out for each person (on the *x*-axis, like for the numeric variable at the *x*-axis of the left panel in Figure 1). If, for example, the predicted wellbeing averaged over all persons decreased with age, we would see a decreasing shape in the partial dependence plot. This would correspond to a monotone main effect of the variable age. Due to the flexibility of random forests to approximate functional forms, it would also be possible that we find a non-monotone effect, such as a u-shaped effect where wellbeing is higher for middle aged and lower for younger and elderly persons.

In corpus linguistics, the rows of a data set typically do not correspond to persons but to linguistic instances, such as occurrences of a verb in a corpus of written or transcribed text. The partial dependence plot is then created accordingly by varying one property of the instance while leaving the other properties as they were

6 Often, these will be the different values of the predictor variable that were observed in the data, but it can also be a grid of values over the range of the predictor variable.

observed, computing the predictions for each instance and then averaging over all instances.

Returning to Figure 1, we saw that the two predictors from this toy example seem to be irrelevant when looking at one predictor at a time, like it is done in a univariate partial dependence plot that marginalizes over all other predictor variables. There are, however, other plots from interpretable machine learning, that can capture the effects of two variables at a time: bivariate partial dependence plots and individual conditional expectation or ICE plots (Goldstein et al. 2015). We will further discuss bivariate partial dependence plots below, but in the following concentrate on introducing ICE plots, because they appear to be less widely known in linguistics.

The main difference between univariate partial dependence plots and ICE plots is that in ICE plots the predictions are not averaged over all observations, but displayed separately for each observation. Considering the above description of how partial dependence plots are constructed, this means that the aggregation in the second step of the procedure is left out, leaving one individual line of predicted values for each observation (i.e., each person or each instance) over the range of the predictor variable. These lines are displayed in Figure 2 for the toy example data already used in Figure 1.

ICE plots thus display the predicted value of the response variable (on the *y*-axis) as a function of the value of a predictor variable (on the *x*-axis) for each observation separately. Accordingly, the partial dependence plots in Figure 1 are the averages over the individual ICE curves in Figure 2. Through this, ICE plots can display more detailed information than partial dependence plots. For more details on partial dependence and ICE plots see Henninger et al. (2023a). We will now describe how ICE plots may help detect interaction effects.

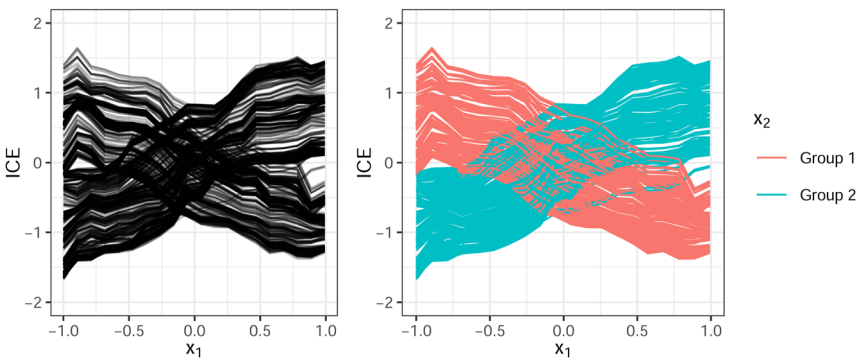

Figure 2: ICE plot for x_1 (left) and ICE plot for x_1 colored w.r.t. x_2 (right).

When we look at the left panel of Figure 2, there seem to be observations for which the effect of the numeric predictor is positive, and others for which it is negative. Moreover, if we color the lines for the individual observations according to the second, binary predictor (see Figure 2, right), we see that the pattern actually corresponds to a perfect interaction between the two predictors. This kind of interaction cannot be detected by looking at one variable at a time (it is the XOR problem mentioned earlier). So univariate partial dependence plots or plots displaying the predicted response only for one predictor at a time are also monofactorial in the sense that they risk to miss interactions.

It is of course also possible to manually include the interaction between two predictor variables as an additional predictor variable in the machine learning model like Gries (2020), and create a partial dependence plot for this additional predictor. The partial dependence plot will then capture the effect of the specified interaction. However, all these approaches do not allow to detect any unexpected and thus unspecified higher order interaction effects: Just like a univariate partial dependence plot cannot capture a 2-factor interaction, a partial dependence plot for a 2-factor interaction, a bivariate partial dependence plot or an ICE plot colored w.r.t. a second variable cannot capture a 3-factor interaction, etc. While we as humans tend to think only in interactions of order two or three at the most, the real pattern in the data may be more complex and contain interactions of higher order.

2.6 Detecting interactions

Gries (2020) suggests an approach based on random forest variable importances that has raised the hope to be able to better identify interactions and has already been picked up in the linguistics literature (Bernaisch and Funke 2024; Deshors 2021; Deshors and Gries 2020; Schmidt and Funke 2024): The first step of this approach is to explicitly add interactions between the predictor variables as additional predictor variables, termed *interaction predictors* in parts of the literature. This approach is perfectly legitimate and is also mentioned in Strobl et al. (2009b). It can increase the prediction accuracy in settings like the XOR case, where predictors have small or no main effects but may exhibit informative interactions, which are easier to detect when explicitly included. However, Gries (2020) suggestion goes beyond this first step and seems to imply that the resulting variable importance scores of the interaction predictors could help identify whether a predictor variable contributes to the prediction through a 2-factor interaction rather than through a main effect.

To illustrate this, Gries (2020) presents a case study with a toy data set. The example has been formulated such that an interaction predictor alone is able to perfectly predict the outcome. The interaction predictors are added and the finding is

described this way (Gries 2020: 639): "Variable importance: every single forest chooses P2:P3 as the by far most important predictor, but it is worth noting that, because of the sampling, all other predictors' variable importance scores are also not 0." This indicates that this result is what the author considers as the correct behavior of the variable importance and to our understanding implies that the approach of including interaction predictors and computing their importance is promoted as a method for detecting truly relevant interaction effects.

Other papers seem to have interpreted Gries' suggestion in the same way. For example, Deshors and Gries (2020: 225) write "[…] we follow Gries's (forthcoming) recommendations: […] the first step of our statistical analysis consisted of manually creating a number of new predictors that essentially represent all two-way interactions […]. These were then added as predictors to a forest […]. […] Second, we computed the version of variable importance scores proposed in Janitza et al. (2013) […]." Deshors and Gries (2020) use a linear surrogate model in addition to the random forest analysis, but concentrate the interpretation on the four interaction predictors that have shown the highest AUC[7] variable importance scores (besides the single variable variety; the variable importance scores for these four interaction predictors are reported on p. 226, the importance scores for all predictors in Appendix B of Deshors and Gries 2020), indicating that the authors consider these interaction effects to be most relevant. A similar approach was followed by Deshors (2021).

Schmidt and Funke (2024, p. 8 of online version) write "Following Gries (2020), we explicitly included interaction variables of the two sociolinguistic variables under investigation, namely GENDER and TIME, with each of the other variables. […]" and conclude (p. 9 of online version) "Figure 1 shows the variable importance scores measured in mean decrease in accuracy. While the main effects of GENDER and TIME rank among the least important variables, the plot highlights the high importance of many of the interaction effects for the model's accuracy. Especially the interaction effects of GENDERxTRIGGER.LEMMA and TIMExTRIGGER.LEMMA turn out to be the most important variables in the random forest followed by the overall effect of TRIGGER.LEMMA and the interaction of TIMExNEWSPAPER." The authors then go on to focus the interpretation on these effects. A similar approach was followed by Bernaisch and Funke (2024).

While it is correct to say that any (interaction) predictors with a high variable importance have highly contributed to predicting the response variable in a random forest, we would like to caution the readers against the idea that including

7 The AUC variable importance of Janitza et al. (2013) was not explicitly investigated in our following simulation study, but is also permutation-based. We therefore expect a similar behavior w.r.t. interaction predictors as for the original permutation importance.

interaction predictors provided a reliable way of detecting whether interaction effects are truly relevant. The evidence from our simulation study, presented below, shows that this is not the case. In particular, we will see that the importance of interaction predictors can be higher than that of individual predictors even when the corresponding interaction effect does not exist while the main effect of the individual predictor does. This means that the practice of concluding from a high variable importance ranking of interaction predictors that the corresponding interaction effect was truly relevant is not justified.

Unlike our simulation study, the case study of Gries (2020) only looks at the ability of the suggested approach to identify an interaction that is actually in the data (true positive finding).[8] In order to validate a new methodological approach, however, it is important to look at different settings, also including ones that allow us to check whether the approach may erroneously identify interactions that are not in the data (false positive findings).

3 Simulation study

In order to more systematically explore these aspects, we have conducted a simulation study. Here we will only present the main findings from this study. Further details are provided in the Appendix A.

For systematically investigating the properties of a new methodological approach, simulation studies have the advantage that one can systematically vary different factors in the model used for generating the data (termed the data generating process) and see how they affect the results. In the simulation study reported here, we varied the pattern of the true effects of the predictor variables and compared the variable importance scores achieved through the approach suggested by Gries (2020) to the true effects of the predictors. In particular, we used two different settings in order to be able to assess both true positive and false positive findings.

In condition A, there were no true interaction effects in the data. Only main effects were simulated for the first three predictor variables (see Figure 3). An ideal

8 As a side note, Gries (2020) claims that the first predictor in his case study was not at all relevant. One can argue against this statement because the first predictor is indeed associated with the response variable, which makes it informative for the prediction from a marginal point of view. The reason for this incongruence is that Gries (2020) seems to implicitly argue from a partial point of view, where the first predictor does not add any information on top of the interaction of the other two predictors. But this is not what the unconditional permutation importance expresses, cf. Debeer and Strobl (2020).

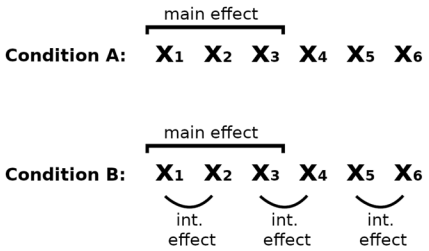

Figure 3: Conditions of the simulation study.

interaction detection method would show high variable importance for the main effects of these predictors, but not for any interaction effects.

In condition B, there were both main effects and two-factor interactions in the data. As can be seen from Figure 3, we included 2-factor interactions where both predictors also have main effects, where only one of the two predictors has a main effect, and where none of the two predictors has a main effect. An ideal interaction detection method would show high variable importance for the true main effects of the first three predictors and also high variable importance for each of the true 2-factor interaction effects.

The results are shown in Figure 4. The figure shows boxplots of the unconditional permutation variable importance scores (over 1,000 simulation repetitions) for each individual predictor (termed here: main effect predictors) and each

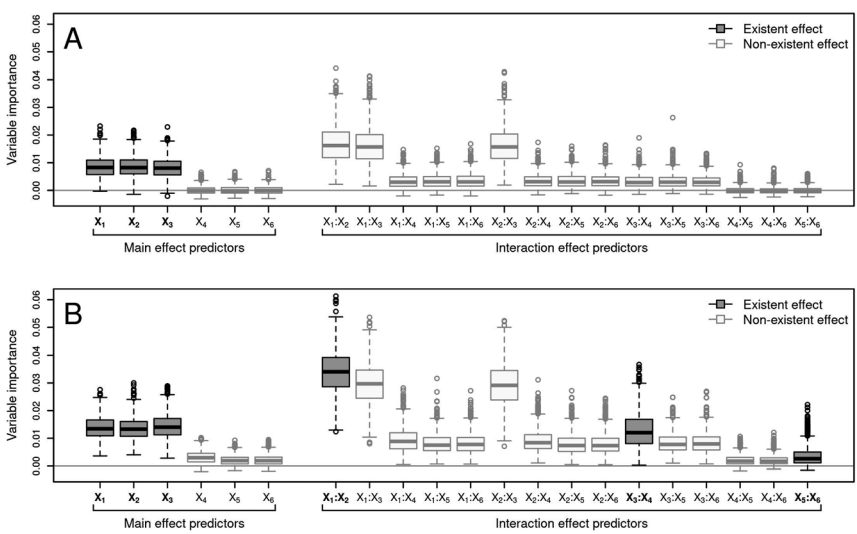

Figure 4: Results of the simulation study.

combination of two predictors (termed here: interaction effect predictors) included in the random forest. The interaction effect predictors were included according to Gries (2020, see Appendix for details).

The boxplots are drawn in black and filled when the respective effect is actually present in the data generating process. So a high average importance in a boxplot drawn in black indicates that the variable importance is a good criterion for identifying an effect that is actually present (true positives). Boxplots drawn in grey and unfilled represent variables or combinations of variables that have no true effect in the data generating process. A high average importance in a boxplot drawn in grey thus indicates that the variable importance has also identified effects as important that are not truly important (false positives).

In the top panel for condition A, we see that the variable importance is increased for the first three variables, which do have main effects (true positives), and centered at zero for the next three variables, which do not have any main effect in condition A (true negatives).

However, the variable importance is also increased for any interaction effect predictors containing one of the first three variables, and even more increased for those containing two of the first three variables (false positives).

Please note that this does not mean that there is something wrong with the variable importance. It actually behaves in the way that is to be expected: The importance for any predictor (combination) containing a particular variable expresses a conglomerate of the effect of that variable alone and in interactions. What it does mean, however, is that comparing the variable importance of main effect predictors and interaction effect predictors is not an appropriate means for identifying true interactions in the data.

This result is further supported when we look at the bottom panel of Figure 4 for condition B. Here we see that, again, the variable importance is increased for the first three variables, which do have main effects (true positives). However, now it is also somewhat increased for the next three variables, which do not have any effect in condition B either (false positive). Again, note that we call this a false positive result only while we investigate the variable importance as a method for distinguishing main effects from interactions. The results show that it is not able to achieve this, because the interaction effects that the next three variables x_4 through x_6 are involved in (see again Figure 3) also carry over to the variable importance of the main effect predictors. This is a legitimate behavior for the variable importance, but rules it out as a method for distinguishing main effects from interactions. Finally, we can see in the rest of the pattern that the variable importance is highest for the interaction effect predictor of the first two variables (which both have a main effect and an interaction; true positive). However, it is second highest for the interaction effect predictors of variables x_1 and x_3, and x_2 and x_3 (which all have main effects, but no

true interaction effects, making them false positives). The interaction effect predictors of variables x_3 and x_4 (true interaction, but only one main effect) and variables x_5 and x_6 (true interaction, but no main effect) show a much lower importance than the pairs with no true interactions but main effects. Again, this shows that the variable importance is not an appropriate means for identifying true interactions in the data.[9]

The variable importance is always a mix of the effects a single variable has in main effects and interactions of different orders, and the same applies to the importance of an interaction predictor. So unfortunately, as appealing as the idea might seem, this is not a valid approach for identifying interactions. It is important to remind ourselves: Random forest and other black box machine learning methods are so good at making predictions because they are so flexible that they can approximate any pattern in the data. If that pattern is complex, any attempt to summarize it in an undercomplex way, such as summarizing a pattern containing 3- and 4-factor interactions with means for detecting main effects and 2-factor interactions only, is bound to fail.

Other approaches that have been suggested for generically identifying interactions in random forests are the approaches of Friedman and Popescu (2008); Hornung and Boulesteix (2021); Ishwaran (2007). Friedman and Popescu (2008) suggest a descriptive interaction statistic for 2- and more-factor interactions between a predictor and any other predictors. Note, however, that this statistic shows false positive results for predictors without main effects (Friedman and Popescu 2008; Henninger et al. 2023a). (Hornung and Boulesteix 2021) suggest so called interaction forests. This approach is quite similar in spirit to the suggestion of Gries (2020), and did show some success detecting interaction effects in empirical data. Also similar to the suggestion of Gries (2020) though, it can produce false positive results (Gitzi 2022). Ishwaran (2007) proposes an approach for identifying 2-factor interactions by means of subtracting the individual permutation importance from that of the joint permutation of both predictors.[10] The evidence from the simulation study of Ishwaran (2007) is rather limited, so more research may be needed to further evaluate this approach. What is conceptually clear, however, is the limitation of any method explicitly specifying 2-factor interactions in the presence of higher order interactions.

9 When assessing the differences between the importance of the interaction effect predictors and the sums of the importance of the respective individual predictors, as suggested by one of the reviewers of this manuscript, the results are inconclusive and do not allow to reliably indicate true effects either.

10 Ishwaran (2007) uses random node assignment instead of random permutation as the basis for his mathematical proofs, but argues that it has similar properties.

So generally speaking, for the time being there is no ideal method for detecting interactions with random forests. However, in corpus linguistics it often seems to be the case that the number of predictor variables is not as high as in many other application areas of random forests, and that many predictor variables are categorical. In this case, it is very well possible to display the predicted responses of one predictor variable separately for all combinations of levels of the other predictor variables to explore higher order interactions. This approach was taken by Szmrecsanyi et al. (2016) (Fig. 3) and Hundt et al. (2020) (Fig. 8) for illustrating 3-factor interactions. This strategy for displaying random forest predictions provides much more information than any single monofactorial plot. It can also be combined with colored ICE plots like those displayed in Figure 2 (right) for one metric and one categorical predictor, or with bivariate partial dependence plots for displaying 2-factor interactions between any two predictor variables. Bivariate partial dependence and ICE plots can be generated, e.g., with the help of the R packages `iml` (Casalicchio et al. 2024; Molnar et al. 2018) and `pdp` (Greenwell 2022, 2017). In order to avoid overinterpretations when visually interpreting multicolored bivariate partial dependence plots, Henninger et al. (2023a) point out that random patterns can be hard to distinguish from true interaction effects for the human eye, and that this can be aggravated by a poor choice of the range of the color scale.

When the number of predictors is too high to display the predicted responses for all combinations of predictor variable levels, however, there is always the possibility that any 2-factor interactions, that can be explicitly specified and investigated, do not tell the whole story, because higher order interactions are involved but remain unspecified. Therefore, exploratory techniques that are able to identify interactions of unknown order would be a valuable extension of our methodological toolbox. A recent approach that looks promising in this regard is the one by Herbinger et al. (2022). This approach searches for clusters of lines in ICE plots that can be explained by one or more additional predictor variables, which corresponds to searching for interactions of order two or higher. However, any approaches based on the selection of variables into a tree or their position within a tree, such as those of Herbinger et al. (2022) and Ishwaran (2007), may be affected by variable selection bias when using traditional greedy search algorithms for tree building.

4 Conclusions

We hope to have been able to clarify a few aspects concerning the interpretation of trees and random forests, as well as some of the technical and statistical aspects surrounding them. We believe that the major strength of tree-based methods is their

flexible, exploratory nature, but the price we pay for their flexibility is that their results are harder to interpret than those of more restrictive, parametric models. On the other hand, parametric models make such strong assumptions about the properties and functional form of the association between predictors and response, that if these assumptions are not correct, we will miss interesting new insights from the data that tree-based methods could have discovered – if we use them in an informed way.

Appendix A: Details on the simulation study

The simulation study was performed using the statistical software R (R Core Team 2023, v4.3.2). To fit random forests based on conditional inference trees, the `cforest` function of the R package `party` (v1.3-13) was used. To calculate (unconditional) permutation importance scores, the `varimp` function of the `party` package was used. Random forests were always created using 1,000 trees (`ntree` = 1,000). In each node of a tree, 5 predictor variables were randomly selected to be evaluated for splitting (`mtry` = 5) which corresponds to the commonly used default of setting `mtry` to the square root of the total number of predictors in a classification setting. The following sections describe the structure of the generated artificial data, the subsequent analysis with random forests and the obtained results of the simulation study.

A.1 Data generating process

One observation of the simulated data always consisted of a (dummy coded) binary response variable (y_i) and six (dummy coded) binary predictor variables (x_{pi}, with p = 1, 2, …, 6). The predictor variables were independently sampled (with $P(X_{pi} = 1) = 0.5$) and were, therefore, not correlated with each other. The generation of the response variable y_i was based on a logistic regression model. The simulation study included two conditions (A and B), which differed from each other in terms of whether the logistic regression equation only included main effects of the predictor variables, or whether interaction effects were included as well. Equation (1) shows the logistic regression equation for condition A.

$$\log\left(\frac{P(Y_i = 1|\mathbf{x}_i)}{P(Y_i = 0|\mathbf{x}_i)}\right) = b_0 + b_1x_{1i} + b_1x_{2i} + b_1x_{3i} \tag{1}$$

with: $\mathbf{x}_i = (x_{1i}, x_{2i}, \ldots, x_{6i})$, $b_0 = -2$ and $b_1 = 1$.

As can be seen in Equation (1), only the first three of the six predictor variables exhibited a main effect, while the other predictors did not contribute to the generation of y_i and no interactions between predictors were present. In addition, the three relevant predictors all exhibited a main effect with the same effect size (b_1).

In condition B, the logistic regression equation was extended with three interaction effects. The added effects were the two-way interactions of the variable pairs (x_1, x_2), (x_3, x_4) and (x_5, x_6). Equation (2) shows the logistic regression equation for condition B.

$$\log\left(\frac{P(Y_i = 1|\mathbf{x}_i)}{P(Y_i = 0|\mathbf{x}_i)}\right) = b_0 + b_1x_{1i} + b_1x_{2i} + b_1x_{3i} + b_1x_{1i}x_{2i} + b_1x_{3i}x_{4i} + b_1x_{5i}x_{6i} \quad (2)$$

with: $\mathbf{x}_i = (x_{1i}, x_{2i}, \ldots, x_{6i})$, $b_0 = -2$ and $b_1 = 1$.

As shown in Equation (2), all main and interaction effects had the same effect size (b_1). It is also evident from Equation (2) that the three added interactions differ in a certain aspect: The first interaction term ($b_1x_{1i}x_{2i}$) is composed of two predictors which both exhibited a main effect as well. The second interaction term ($b_1x_{3i}x_{4i}$) consists of two predictors of which one exhibited a main effect while the other did not. And the third interaction term ($b_1x_{5i}x_{6i}$) consists of two predictors which both did not exhibit a main effect. Figure 3 in the main text visualizes the patterns of the present main and interaction effects in the two simulation conditions.

A.2 Random forest analysis

In each replication of the simulation study, an artificial data set consisting of 500 observations was created according to the data generating process described above. According to the proposal by Gries (2020), the data set was extended by adding new predictor variables, which are supposed to represent the combined effect of two predictors. Specifically, each newly added variable was a four-level categorical variable, representing the four possible combinations of two binary predictors.[11] For example, the newly added variable x_1:x_2 represented the four possible combinations of the two binary predictors x_1 and x_2 (see Table 1).

For the six predictor variables (x_1, x_2, …, x_6) in our artificial data, this approach resulted in the addition of 15 new variables (x_1:x_2, x_1:x_3, …, x_5:x_6), one for each predictor pair. The approach proposed by Gries (2020) then proceeded with fitting a

11 Including all four combinations of the two factor levels of two binary predictors is not how an interaction is typically encoded in statistical models. In linear models, interaction effects are typically encoded as products of two binary predictors. While the results presented here stick to the proposal by Gries (2020) with four categories, a more extensive simulation study by Theiler (2021) has also included the product encoding. The results show the same pattern as the results presented here.

Table 1: Possible values of the categorical variable x_1:x_2.

x_1	x_2	x_1:x_2
0	0	0.0
0	1	0.1
1	0	1.0
1	1	1.1

random forest to the extended data, using both the original predictor variables and the newly created variables. Then the (unconditional) permutation importance scores were calculated for all predictor variables. In total, 1,000 replications were run for each condition.

Given the intention of the approach proposed by Gries (2020) to use the variable importances of the newly added predictors to identify interaction effects, we refer to the binary predictors (x_1, x_2, …, x_6) as "main effect predictors" and to the four-level predictors (x_1:x_2, x_1:x_3, …, x_5:x_6) as "interaction effect predictors" in the following.

A.3 Results

Figure 4 in the main text shows the distribution of the variable importance scores for each predictor over the simulation replications in the two conditions of the simulation study. The respective boxplots are drawn in black and filled when an effect is actually present in the data generating process. So a high average importance in a boxplot drawn in black indicates that the variable importance is a good criterion for identifying an effect that is actually present (true positives). Boxplots drawn in grey and unfilled represent variables or combinations of variables that have no true effect in the data generating process. A high average importance in a boxplot drawn in grey thus indicates that the variable importance has also identified effects as important that are not truly important (false positives). For a more detailed interpretation please refer to the main text.

The more extensive simulation study by Theiler (2021) also included a setting with 3-factor interactions. The approach proposed by Gries (2020) also showed false positive results in this setting. This further supports the conclusion in the main text that random forest variable importance scores are a mix of the effects a predictor variable has in main effects and interactions of different orders and are thus not suited for identifying interactions.

References

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

Bernaisch, Tobias. 2022. *Comparing generalised linear mixed-effects models, generalised linear mixed-effects model trees and random forests: Filled and unfilled pauses in varieties of English*, 163–193. Cambridge, England: Cambridge University Press.

Bernaisch, Tobias & Nina Funke. 2024. Particle placement in Hong Kong English: Independence from great Britain as a trigger of structural change? *Journal of English Linguistics* 52(2). 137–163.

Breiman, Leo. 1996. Bagging predictors. *Machine Learning* 24(2). 123–140.

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

Breiman, Leo, Jerome H. Friedman, Richard A. Olshen & Charles J. Stone. 1984. *Classification and regression trees*. New York: Chapman & Hall.

Casalicchio, Giuseppe, Christoph Molnar & Patrick Schratz. 2024. iml: Interpretable machine learning. R package.

Debeer, Dries & Carolin Strobl. 2020. Conditional permutation importance revisited. *BMC Bioinformatics* 21(1). 307.

Debeer, Dries, Torsten Hothorn & Carolin Strobl. 2021. permimp: Conditional permutation importance. R package.

Deshors, Sandra C. 2021. Contextualizing past tenses in L2: Combined effects and interactions in the present perfect versus simple past alternation. *Applied Linguistics* 42(2). 269–291.

Deshors, Sandra C. & Stefan Th. Gries. 2020. Mandative subjunctive versus should in world Englishes: A new take on an old alternation. *Corpora* 15(2). 213–241.

Friedman, Jerome H. & Bogdan E. Popescu. 2008. Predictive learning via rule ensembles. *Annals of Applied Statistics* 2(3). 916–954.

Gitzi, Valerie. 2022. Interaktionen im Interaction Forest. Master's thesis. Author's University.

Goldstein, Alex, Adam Kapelner, Justin Bleich & Emil Pitkin. 2015. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational & Graphical Statistics* 24. 44–65.

Greenwell, Brandon M. 2017. pdp: An R package for constructing partial dependence plots. *The R Journal* 9(1). 421–436.

Greenwell, Brandon M. 2022. pdp: Partial dependence plots. R package.

Gries, Stefan Th. 2020. On classification trees and random forests in corpus linguistics: Some words of caution and suggestions for improvement. *Corpus Linguistics and Linguistic Theory* 16(3). 617–647.

Gries, Stefan Th. 2021. *Statistics for linguistics with R*. Berlin, Boston: De Gruyter Mouton.

Henninger, Mirka, Rudolf Debelak, Yannick Rothacher & Carolin Strobl. 2023a. Interpretable machine learning for psychological research: Opportunities and pitfalls. *Psychological Methods.* Advance Online Publication. https://doi.org/10.1037/met0000560.

Henninger, Mirka, Rudolf Debelak & Carolin Strobl. 2023b. A new stopping criterion for Rasch trees based on the Mantel-Haenszel effect size measure for differential item functioning. *Educational and Psychological Measurement* 83(1). 181–212.

Herbinger, Julia, Bernd Bischl & Giuseppe Casalicchio. 2022. REPID: Regional effect plots with implicit interaction detection. In *Proceedings of the 25th international conference on artificial Intelligence and statistics*, vol. 151, 10209–10233.

Hothorn, Torsten & Achim Zeileis. 2015. partykit: A modular toolkit for recursive partytioning in R. *Journal of Machine Learning Research* 16(118). 3905–3909.

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

Hornung, Roman & Anne-Laure Boulesteix. 2021. Interaction forests: Identifying and exploiting interpretable quantitative and qualitative interaction effects. *Computational Statistics & Data Analysis* 171(8). 107460.

Hothorn, Torsten, Kurt Hornik, Carolin Strobl & Achim Zeileis. 2024a. party: A laboratory for recursive partytioning. R package.

Hothorn, Torsten, Heidi Seibold & Achim Zeileis. 2024b. partykit: A toolkit for recursive partytioning. R package.

Hundt, Marianne, Paula Rautionaho & Carolin Strobl. 2020. Progressive or simple? A corpus-based study of aspect in world Englishes. *Corpora* 15(1). 77–106.

Ishwaran, Hemant. 2007. Variable importance in binary regression trees and forests. *Electronic Journal of Statistics* 1. 519–537.

Janitza, Silke, Carolin Strobl & Anne-Laure Boulesteix. 2013. An AUC-based permutation variable importance measure for random forests. *BMC Bioinformatics* 14. 119.

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

Kuhn, Max. 2008. Building predictive models in R using the caret package. *Journal of Statistical Software* 28(5). 1–26.

Kuhn, Max, Jed Wing, Steve Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, Zachary Mayer, Brenton Kenkel, Michael Benesty, Reynald Lescarbeau, Andrew Ziem, Luca Scrucca, Yuan Tang, Can Candan, Tyler Hunt & R Core Team. 2023. caret: Classification and regression training. R package.

Liaw, Andy & Matthew Wiener. 2022. randomForest: Breiman and Cutler's random forests for classification and regression. R package; Fortran original by Leo Breiman and Adele Cutler.

Loh, Wei-Yin & Yu-Shan Shih. 1997. Split selection methods for classification trees. *Statistica Sinica* 7(4). 815–840.

Molnar, Christoph, Bernd Bischl & Giuseppe Casalicchio. 2018. iml: An R package for interpretable machine learning. *JOSS* 3(26). 786.

Philipp, Michel, Achim Zeileis & Carolin Strobl. 2016. A toolkit for stability assessment of tree-based learners. In Ana Colubi, Angela Blanco & Cristian Gatu (eds.), *Proceedings of COMPSTAT 2016 – 22nd international conference on computational statistics*, 315–325. Oviedo: The International Statistical Institute/International Association for Statistical Computing.

Philipp, Michel, Thomas Rusch, Kurt Hornik & Carolin Strobl. 2018. Measuring the stability of results from supervised statistical learning. *Journal of Computational & Graphical Statistics* 27(4). 685–700.

Philipp, Michel, Carolin Strobl, Achim Zeilei, Thomas Rusch, Kurt Hornik & Lennart Schneider. 2023. stablelearner: Stability assessment of statistical learning methods. R package.

Quinlan, J. Ross. 1986. Induction of decision trees. *Machine Learning* 1(1). 81–106.

Quinlan, J. Ross. 1993. *C4.5: Programms for machine learning*. San Francisco: Morgan Kaufmann Publishers Inc.

R Core Team. 2023. *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing.

Rothacher, Yannick & Carolin Strobl. 2023. Identifying informative predictor variables with random forests. *Journal of Educational and Behavioral Statistics* 49(4). 595–629.

Schmidt, Karola & Nina Funke. 2024. Exploration of the mandative subjunctive in Pakistani English. *World Englishes*. Advance Online Access. https://doi.org/10.1111/weng.12697.

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

Strobl, Carolin, Anne-Laure Boulesteix, Achim Zeileis & Torsten Hothorn. 2007b. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics* 8. 25.

Strobl, Carolin, Anne-Laure Boulesteix, Thomas Kneib, Thomas Augustin & Achim Zeileis. 2008. Conditional variable importance for random forests. *BMC Bioinformatics* 9(307). 1471–2105.

Strobl, Carolin, Torsten Hothorn & Achim Zeileis. 2009a. Party on! A new, conditional variable importance measure for random forests available in the party package. *The R Journal* 1(2). 14–17.

Strobl, Carolin, James Malley & Gerhard Tutz. 2009b. An introduction to recursive partitioning: Rationale, application and characteristics of classification and regression trees, bagging and random forests. *Psychological Methods* 14(4). 323–348.

Szmrecsanyi, Benedikt, Jason Grafmiller, Benedikt Heller & Melanie Röthlisberger. 2016. Around the world in three alternations: Modeling syntactic variation in varieties of English. *English World-Wide* 37(2). 109–137.

Tagliamonte, Sali A. & R. Harald Baayen. 2012. Models, forests, and trees of York English: Was/were variation as a case study for statistical practice. *Language Variation and Change* 24(2). 135–178.

Theiler, Sven. 2021. Detektion von Interaktionen in Random Forests. Master's thesis. Author's University.

White, Allan P. & Wei-Zhong Liu. 1994. Bias in information based measures in decision tree induction. *Machine Learning* 15(3). 321–329.