Permuting the out-of-bag values versus randomizing node assignments when measuring variable importance in random forests: A comparison study

MASTER'S THESIS

Author: Katrin Racic-Rachinsky

Supervisor: Dr. Roman Hornung

Institut für Statistik
Ludwig-Maximilians-Universität München
December 15, 2021

Abstract

The random forest algorithm is popular because it can deal with high-dimensional data, complex interactions and highly correlated covariates. Another advantage is that it offers built-in variable importance measures (VIM) that identify relevant variables, provide variable rankings and can also be used for variable selection. The most well-known and popular VIM is the permutation accuracy importance, but it has no straightforward way to be applied to data with missing values, which motivated Hapfelmeier et al. (2014) to introduce a new VIM. The idea of this new approach is to break the association between a covariate and the response by randomly assigning observations to nodes at all splits that depend on the variable of interest instead of permuting the values of the variable of interest. It could be shown that this new approach provides sensible results while being well able to deal with missing values.

The aim of this thesis is to compare by means of a simulation study the performance of both methods, the original permutation importance and the new approach, in situations with no missing data. Both VIMs are evaluated by the correlation strength between importance values and effect sizes, or in the case of constant effect sizes, by measuring the ability to discriminate between relevant and irrelevant covariates with the AUC. Different effect sizes and parameter settings are considered for a binary response and normally distributed covariates.

The results of the simulation study show that both methods achieve a very similar performance in all considered simulation settings. For most settings, no significant difference between the performances could be detected. Of the few significant differences that could be found, however, almost all are in favour of the new approach, but the differences between the performances of both measures are in all cases very small.

Contents

List of Figures

List of Tables

1 Introduction

Random Forest, introduced by Leo Breiman in 2001, is a nonparametric ensemble method for classification or regression that is based on the concept of decision trees. Random forests are popular and widely used because they show high prediction accuracy and are able to handle high-dimensional data, complex interactions and highly correlated covariates. Since they use bootstrapping or subsampling to construct the trees of the ensemble, there is a so-called out-of-bag (OOB) sample for each tree, i.e. the part of the original data that was not used to build the tree, which can be used as test data. Moreover, with built-in variable importance measures, they can identify influential covariates, considering not only the individual influence of a covariate, but also its impact in multivariate interactions with other covariates. (cf. Strobl et al., 2008)

Such a variable importance measure ranks the covariates according to their relevance and can also be used for variable selection. The most popular and well-established variable importance measure is the permutation variable importance. It evaluates the impact of a covariate by assessing how much the OOB-accuracy decreases when the association between that covariate and the response is broken by permuting the values of that covariate in the OOB-sample. Random forests themselves can handle missing data very easily by the use of surrogate splits, but the permutation importance has no straightforward way to handle missing values, which means you have to resort to either imputation methods or to the exclusion of observations (complete case analysis). This shortcoming of the permutation importance prompted Hapfelmeier et al. (2014) to introduce a new variable importance measure that solves this issue. This new approach is actually very similar to the permutation importance, but instead of permuting the values of the covariate, it breaks its association to the response by randomly assigning observations to nodes in splits that depend on the covariate of interest, in the same proportion in which the observations were originally sorted into the nodes. In doing that, the assignment of observations to nodes is detached from the raw values of the covariate, so that computing importance values in the presence of missing values is no longer a problem.

The new approach has been shown to be superior to the permutation importance in terms of being able to deal with missing data, but the question is whether is has further advantages or disadvantages over the permutation importance. To learn more about possible differences and similarities between the methods, their performance is compared in this thesis by means of a simulation study with no missing values in the simulated data. First, Section 2 gives a brief overview about the random forest algorithm and the methods on which it is based before introducing the conditional inference framework developed by Hothorn et al. (2006). This framework is the basis of the R package party, which is used in this thesis to construct random forests and compute variable importances. Section 3 outlines the concepts of both the permutation accuracy importance and the new variable importance measure introduced by Hapfelmeier et al. (2014) and gives information about differences between both methods. In Section 4, the design of the simulation study is explained, including the data generating process and the parameter settings, and the way in which the performance of the importance measures is evaluated is explicated. The last part of the section contains an overview over all scenarios of the simulation study. In Section 5 the results of the simulation study are presented. The performance of both methods is analysed first for the scenarios with variating effect sizes, and then for the scenarios with constant effect sizes, before the variable importance values themselves are compared for both importance measures. In the last part of Section 5, the results are briefly summarized, before Section 6 concludes the thesis.

2 Random Forest

2.1 The Random Forest algorithm

Random forest is a machine learning algorithm that can be used both for classification and regression tasks. It uses a collection of decision trees that are built from bootstrap samples and makes predictions by taking the average of the predictions of the individual trees in case of a regression task, or the majority vote of the predictions of the trees in case of a classification task.

The decision trees themselves are based on the principle of recursive partitioning: the data is split into binary subsets (nodes), which are again repeatedly split into binary subsets. In that way, the data, and more generally the feature space, is again and again partitioned into increasingly smaller nodes until some stopping criterion is met, for example a minimal node size. For each terminal node, a constant response can be estimated from the data within this node by taking simply the average of the response values (for a continuous response) or the category with the highest proportion (for a categorical response). The aim in splitting a node is to reduce the 'impurity' within nodes, i.e. child nodes should have within themselves less variability concerning the response than their parent node had. The split is conducted by selecting one of the covariates, called then the *split variable*, and a split point for that covariate that separates the values of that covariates into two disjunct subsets. All observations are then sorted into one of either child node, dependent on their value of the split variable. (cf. Breiman et al., 1984)

A random forest usually employs the principle of *bagging* (cf. Breiman, 1996), which is short for 'bootstrap aggregating' and means in general that bootstrap samples are used to create an ensemble of models, whose predictions are then aggregated. In case of a random forest, each tree of the ensemble is built from a bootstrap sample from the original data, i.e. if we have training data of size n, for each tree we randomly draw with replacement a sample of size n from that data. It is also possible to use subsampling instead of bootstrapping, which means that the samples for the individual trees are drawn without replacement and their size is just a fraction of the original training data. A typical fraction used is 0.632, as this corresponds approximately to the fraction of unique observations that are on average drawn into a bootstrap sample (cf. Strobl et al., 2007), though of course other fractions can be used.

Due to the use of bootstrap sampling or subsampling, there are for each tree in a random forest some observations from the original data that were not drawn into the sample that was used to construct this tree. These observations are called out-of-bag (OOB) observations. They serve as an built-in test set for the respective tree, and by averaging the out-of-bag predictions of the individual trees, the error of these predictions can be used as an estimate of the generalization error of the forest. (cf. Cutler et al., 2011)

Trees are approximately unbiased, but unstable due to their hierarchical structure, where one small change in one of the top splits effects all splits below and therefore the complete tree structure and prediction. That makes them ideal candidates for bagging, which brings the most improvement in performance for methods with a high variance and a low bias. However, simple tree bagging has the disadvantage that the individual trees of the ensemble are correlated since they all come from very similar data. In order to reduce that correlation, random forests add another idea to the procedure of bagging: for each split only a subset of covariates of size $mtry < p$ (p being the total number of covariates) is randomly chosen to be considered for the split. In consequence, though all trees of the ensemble are still built from very similar data, they differ much more in their choices of splits and are therefore less correlated to each other. Reducing the correlation between trees reduces the variance of the average prediction of all trees, as shown in chapter 15.2 of Hastie et al. (2009).

Moreover, with the random restriction of the set of potential split variables, weaker covariates may be selected that would not have been chosen if all p covariates had been available for the split. That does not seem to be an advantage at first glance, but splits that may not be locally optimal in the sense that they lead to less reduction of impurity than other splits can still improve the overall performance of a tree, since the algorithm for growing a tree is greedy in the way that it always chooses the locally best split without taking into account the effects which that split has on following splits. Through the selection of a weaker covariate, interaction effects may be detected that would have been missed otherwise. (cf. Strobl et al., 2008, Strobl et al., 2009)

Common default values for $mtry$ are $\lfloor \sqrt{p} \rfloor$ for classification and $\lfloor \frac{p}{3} \rfloor$ for regression, but dependent of the individual data situation using other values might improve the performance of a random forest. In general, random forests are not very sensitive to changes in their parameter settings, they mostly offer a good performance "off the shelf"and benefit less from hyperparameter tuning than other machine learning algorithms. Among the hyperparameters of a random forest, however, $mtry$ is usually the one that is the most tunable. (cf. Cutler et al., 2011; Probst et al., 2018)

Like other tree-based models, random forests have a very intuitive way to handle missing values: while observations containing missing values in a potential split variable are simply ignored for the computation of the impurity reduction, so-called surrogate splits are used to decide which child node observations are assigned to when they have a missing value in the selected split variable. Surrogate splits mean that another covariate and corresponding split point is used, that partitions the data in a very similar way to that of the selected split variable. Several surrogate splits can be computed and ranked according to how good they mimic the original split. Whenever there is a missing value in the split variable, the first-ranked surrogate split can be used, and if it also has a missing value in that observation, the second-ranked surrogate split can be used, and so forth. (cf. Strobl et al., 2009; chapter 9.2.4 in Hastie et al., 2009)

2.2 Selection bias and Conditional Inference Forests

The original random forest algorithm, as it was introduced by Leo Breiman in 2001, consists of trees that are constructed with the CART algorithm (cf. Breiman et al., 1984), which means all possible splits over all covariates are considered and that one is chosen which leads to the highest decrease in impurity, measured by the Gini index in case of a categorical response, and by the mean squared error in case of a continuous response. It has been shown though, that if splits are selected based on the decrease in Gini impurity, continuous covariates and those with many categories are more likely to be chosen than covariates with few categories. Moreover, this splitting method is also biased towards selecting variables with lots of missing values. (cf. Strobl et al., 2007)

Several ideas have been introduced to prevent that selection bias in trees (cf. e.g. White and Liu, 1994; Loh and Shih, 1997; Dobra and Gehrke, 2001), among which is the concept of *conditional inference trees* (cf. Hothorn et al., 2006). In the conditional inference algorithm, the selection of the covariate used for the split is separated from the selection of the split point of the already selected split variable: instead of searching through all possible splits in all covariates, only the covariate for splitting is chosen in a first step, and then in a second step, the best split for that covariate is selected. The algorithm is implemented both for single trees and for ensembles of trees, which are called *conditional inference forests* (CIFs), in the R package `party`, which is used for all analyses in this thesis.

In a CIF, for training data with n observations, every node of a tree is represented by a vector of case weights $\boldsymbol{w} = (w_1,\ w_2, \ldots,\ w_n)$, in which all observations that are elements of the node are represented by a non-zero weight w_i and the case weights of all observations that are not in the node are set to zero. In the first step of the algorithm, the global null hypothesis of independence between any of the *mtry* covariates and the response is tested for the weight vector $\boldsymbol{w}$ using a permutation test. If the null hypothesis cannot be rejected on a pre-specified significance level α, the node will not be divided, which means that the parameter α is also a hyperparameter determining the tree size. If the null hypothesis is rejected, the covariate with the strongest association to the response (measured by test statistics or p-values) is selected for the split. (cf. Hothorn et al., 2006)

In the second step, all possible binary partitions for the split variable selected in step one are considered to find the optimal split point. The response values of the observations with case weights > 0 (i.e. those within the node that is supposed to be split) are, for each possible binary partition, separated into two samples according to whether the corresponding values of the selected covariate are in one or the other of the binary subsets. The discrepancy between these two samples is measured with a linear statistic, and the split is chosen for which a test statistic based on this two-sample linear statistic is maximal. Hothorn et al. (2006) also state that alternatively any other splitting rule, e.g. that of the CART-algorithm, or even one that allows multiway splits, could be applied to establish the split after having selected the split variable in step one.

3 Variable importance measures

3.1 Variable importance in random forests

One of the advantages of random forests is that they also offer a way to measure the variable importance (VI), i.e. the impact each individual covariate has on the response. A random forest therefore does not only provide reliable predictions but it is also able to give information about which variables contribute the most to the algorithm, so that the random forest algorithm is not a complete black-box model but more interpretable. Besides, such a variable importance measure (VIM) can also be used for variable selection, which is especially helpful in high-dimensional data settings.

One way to measure the variable importance of a variable X_j is to simply sum up in each tree the decrease in Gini impurity in the splits where this variable was used, and average this decrease in impurity over all trees. A variable that is never chosen for any split in any of the trees consequently has a variable importance of zero, while variables tend to get higher importance scores the more often they are selected for splits. Being based on the Gini index, this *Gini importance* also suffers from the selection bias mentioned in Section 2.2, and though it is often available in random forest implementations, for example in the `R` packages `randomForest` and `ranger`, it is not considered further in this thesis.

3.2 Permutation variable importance

A very popular variable importance measure is the *permutation accuracy importance*. The idea is to asses how much worse the prediction accuracy is, evaluated on the OOB observations, if the (potential) association between the covariate of interest and the response is destroyed by permuting the values of the covariate of interest in the OOB sample. The more relevant the covariate is, the higher difference between the original prediction accuracy and the prediction accuracy after the permutation can be expected. The procedure is simple: for each tree, the OOB accuracy is computed both before and after permuting the values of the covariate of interest, X_j, in the OOB sample, then the difference between these accuracies is calculated. This difference can be interpreted as the importance of the covariate X_j in this tree. The average of all these tree variable importances of X_j over the whole ensemble constitutes the final variable importance score of the covariate X_j. (cf. Hapfelmeier et al., 2014)

More precisely, the calculation of the permutation importance can be formalized as follows: Let $\overline{\mathcal{B}}^t$ denote the OOB sample for a bootstrap sample $\mathcal{B}^t$ for a tree t, with $t \in \{1, \ldots, ntree\}$ ($ntree$ being the total number of trees in the ensemble). Then the importance of a covariate X_j in tree t is

$$VI_t(X_j) = \frac{\sum\limits_{i \in \overline{\mathcal{B}}^t} \mathbb{I}(y_i = \hat{f}^t(\boldsymbol{x}_i))}{|\overline{\mathcal{B}}^t|} - \frac{\sum\limits_{i \in \overline{\mathcal{B}}^t} \mathbb{I}(y_i = \hat{f}^t(\boldsymbol{x}_i^{perm_j}))}{|\overline{\mathcal{B}}^t|}, \tag{1}$$

where $\hat{f}^t(\boldsymbol{x}_i)$ denotes the prediction made by tree t for the i-th observation with the original, unpermuted covariate X_j, while $\hat{f}^t(\boldsymbol{x}_i^{perm_j})$ denotes the prediction made by tree t for the i-th observation after the permutation of the values of the covariate X_j, and $\mathbb{I}(\cdot)$ denotes the indicator function. The variable importance for covariate X_j is then computed by taking the average importance over all trees:

$$VI(X_j) = \frac{1}{ntree} \sum_{t=1}^{ntree} VI_t(X_j) \tag{2}$$

(this formalization, slightly changed, is borrowed from Strobl et al., (2008))

Obviously, $VI_t(X_j) = 0$ by definition if the covariate X_j is not selected for any split in tree t. For covariates that are not associated with the response and were nevertheless by chance selected for a split, VI_t, and therefore also the overall variable importance, can even take (small) negative values, if the permuted covariate values accidentally lead to an increase in prediction accuracy (cf. Strobl et al., 2009). Of course, in the same way the permutation importance of a completely irrelevant covariate can also take small positive values when the permuted covariate values by chance lead to a slight improvement in prediction accuracy. So, variable importance values near zero signify that the covariate has no influence on the response, while large values indicate a strong association between the covariate and the response. Since there is no clear rule, it can be difficult though to decide where to set the boundary to discriminate between relevant and irrelevant covariates. This issue has been approached for example by Janitza et al. (2018).

3.3 A new approach by Hapfelmeier et al.

Hapfelmeier et al. (2014) state that one drawback of the permutation importance is that its computation and interpretation is not clear when the data contains missing values, and especially when surrogate splits are used to compute the OOB accuracy. This is their main motivation to suggest a new approach for assessing the variable importance in a random forest. It follows the main idea of the permutation importance, namely to calculate the OOB prediction accuracy after breaking the (potential) association between covariate and response and to compare it to the original OOB prediction accuracy. The difference, however, lies in how the association is broken: instead of permuting the values of the split variable, the observations are randomly assigned to one of the child nodes in the same proportions as they were in the original split. In consequence, the decision which observation goes into what node no longer depends on the raw values of the split variable, which means that handling missing values and the use of surrogate splits is no longer a problem. (cf. Hapfelmeier et al., 2014)

The procedure can be formalized in the following way: Let D be a binary random variable that indicates whether an observation goes into the left ($D = 0$) or right ($D = 1$) child node. For a node k, the probability of sending an observation to one of the child nodes is denoted by $P_k(D = 0)$ and $P_k(D = 1) = 1 - P_k(D = 0)$, respectively. Under the null hypothesis that it does not depend on a covariate X_j which observations go into which child node, it holds that $P_k(D|X_j) = P_k(D)$, so whether there are any missing values in X_j is of no consequence for the decision how an observation is processed down the tree. (cf. Hapfelmeier et al., 2014)

To compute the OOB accuracy, the relative frequency $\hat{p}_k(D = 0) = n_{k,left}/n_k$ replaces the probability $P_k(D = 0)$, where n_k is the number of observations in the parent node k and $n_{k,left}$ is the number of observations in the left child node of node k. The only difference to the original permutation importance is that at each split at a node k where X_j is the split variable, the observations are each randomly assigned with $\hat{p}_k(D = 0)$ to the child nodes of node k, instead of processing the observations down the trees with permuted values of the covariate X_j. (cf. Hapfelmeier et al., 2014) In the following, the new procedure by Hapfelmeier et al. will be denoted by *PropRandom*, short for *proportional randomisation*, referring to Hornung and Boulesteix (2021), who coined that term.

3.4 Possible advantages of the new approach

Hapfelmeier et al. (2014) show, with a simulation study and an application to real data, that their new approach is indeed well suited to deal with missing values and does not artificially inflate the importance values of covariates with missing values. Since the permutation importance cannot be computed for data with missing values, common strategies are to either conduct a complete case analysis, i.e. to exclude all observations that contain missing values, or to use imputation. A complete case analysis, however, often means a substantial loss of information, and it may induce biased inference when values are not missing completely at random. According to the results of Hapfelmeier et al. (2012), a complete case analysis arbitrarily decreases the importance of covariates that were completely observed. These disadvantages of complete case analyses indicate that using PropRandom for data with missing values possibly provides a more reliable and meaningful variable ranking. As for imputation methods, most notably *multiple imputation by chained equations* (MICE, cf. White et al., 2011), they take another approach to the issue of missing values: imputation methods aim at restoring the information that is missing, i.e. they try to simulate a complete data set, while "the rationale of [PropRandom] is not to undo the influence missing values have on the information carried by a variable [...] but to reflect the remaining information that the variable has with the respective values missing." (Hapfelmeier et al., 2014, p. 33)

Besides the straightforward handling of missing values, PropRandom might also discriminate better between relevant and less relevant covariates than the permutation variable importance. The explanation, as outlined by Hornung and Boulesteix (2021, Supplementary Material 1/C) is as follows: The covariate variable spaces of the nodes become less and less general in the lower layers of the trees, as the observations are again and again split into ever smaller subsets. In the root node, a covariate follows an unconditional distribution, while in nodes further down the tree, the distribution of a covariate is conditional on all the splits made before in that branch of the tree. With the permutation importance, however, the values of the variable of interest of the whole OOB-sample are permuted, which means that the permuted values follow the unconditional distribution of the variable in the root node, which is not the same as the conditional distributions in the lower layers of the tree. In consequence, when the permuted values are used for a split in the lower layers of a tree, they have possibly a very different range from the values of the unpermuted variable at this split point, and in that case most of the observations or even all are assigned to one child node, so that the sizes of the two child nodes differ widely from the sizes they have when using the unpermuted variable. (cf. Hornung and Boulesteix, 2021, Supp. Mat. 1/C)

This problem becomes the more pronounced the further down the tree we move, while in the upper layers the ranges of the original values and the permuted values are still quite similar so that the observations with the permuted covariate values will be assigned to the respective child nodes in similar proportions as the observations before the permutation. Accordingly, permuting the values of a covariate brings a smaller decrease in accuracy when that covariate is selected as split variable in the upper layers of a tree than when it is selected in the lower tree layers. Since stronger covariates are more likely to be selected as split variables in the upper layers while weaker covariates are more likely to be selected further down the tree, the permutation importance might attribute too large values to weak covariates. In other words, the permutation accuracy might tend to not separate weaker and stronger covariates well enough. (cf. Hornung and Boulesteix, 2021, Supp. Mat. 1/C)

The PropRandom importance, on the other hand is not affected by the issue described above, since the observations are always assigned to nodes in the same proportions as when the original accuracy was computed and the assignment of observations to nodes is detached from the values of the covariate. To assess whether there is a relevant difference between the two variable importance measures because of the suspected weaker discriminative ability of the permutation importance, Hornung and Boulesteix (2021, Supp. Mat. 1/C) compared the skewness of the distributions of the importance values of both methods in 214 data sets, since a good separation between important and less important covariates would lead to a large skewness value. For 71% of the data sets, the skewness estimates were larger for the PropRandom-based importance values than for the permutation importance values, and Wilcoxon signed-rank tests for paired data showed that the differences were highly significant. To exclude the possibility that the differences were caused by discrepancies in the variable ranking of the two methods, correlations between the importance values of both methods were computed and showed that the rankings were indeed very similar. (For more details, cf. Hornung and Boulesteix, 2021, Supp. Mat. 1/C)

4 Simulation study design

4.1 Data generating process

The simulated data sets consist of 80 covariates, of which only 30 are influential, and a binary response Y, which was modeled by means of a logistic model:

$$P(Y = 1|X = \boldsymbol{x}) = \frac{e^{\boldsymbol{x}^\intercal \boldsymbol{\beta}}}{1 + e^{\boldsymbol{x}^\intercal \boldsymbol{\beta}}} \ ,$$

with effect vector $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \ldots, \beta_{80})^\intercal$. Since 50 of the covariates should be mere noise variables, the last 50 entries of the effect vector are set to zero: $\beta_{31} = \beta_{32} = \cdots = \beta_{80} = 0$.

To create the simulated data, n observations are drawn from a multivariate normal distribution with mean vector $\boldsymbol{\mu} = \mathbf{0}$ and a covariance matrix

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{A} & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \boldsymbol{A} & 0 & 0 & 0 & 0 & & \\ 0 & 0 & \boldsymbol{A} & 0 & 0 & 0 & & \\ 0 & 0 & 0 & \boldsymbol{A} & 0 & 0 & & \vdots \\ 0 & 0 & 0 & 0 & \boldsymbol{A} & 0 & & \\ 0 & 0 & 0 & 0 & 0 & 1 & & \\ \vdots & & & & & & \ddots & \\ 0 & & & \cdots & & & & 1 \end{pmatrix},$$

with

$$\boldsymbol{A} = \begin{pmatrix} 1 & 0 & \rho & 0 & \rho & 0 \\ 0 & 1 & 0 & \rho & 0 & \rho \\ \rho & 0 & 1 & 0 & \rho & 0 \\ 0 & \rho & 0 & 1 & 0 & \rho \\ \rho & 0 & \rho & 0 & 1 & 0 \\ 0 & \rho & 0 & \rho & 0 & 1 \end{pmatrix},$$

so that all covariates have the same variance of 1, which means the covariance matrix is the same as the correlation matrix, and ρ is not only the covariance but also the correlation between covariates. All 30 influential covariates are correlated in blocks of three with a constant correlation of strength ρ, while the 50 noise variables are not correlated to any other variables. The structure of the covariance matrix is as depicted above with an interlocked blockwise correlation (i.e. covariates X_1, X_3, X_5 are correlated to each other and covariates X_2,

X_4, X_6 are correlated to each other, etc.), because the effects are alternatingly positive and negative and it does not make sense to have a positive correlation between a positive and a negative effect. ρ was set to 0 and 0.5, respectively, meaning that in the first case there is no correlation at all between covariates. For the size of the data sets, $n = n_1 = 200$ and $n = n_2 = 1000$ was chosen, so that both small and moderate-sized data sets are considered in the simulation study.

Four alternative choices of effects are considered in the simulation study. These variations of $\boldsymbol{\beta}$ are denoted in the following as $\boldsymbol{\beta}_I = (\beta_{1,I}, \ldots, \beta_{80,I})^\intercal$, $\boldsymbol{\beta}_{II} = (\beta_{1,II}, \ldots, \beta_{80,II})^\intercal$, $\boldsymbol{\beta}_{III} = (\beta_{1,III}, \ldots, \beta_{80,III})^\intercal$ and $\boldsymbol{\beta}_{IV} = (\beta_{1,IV}, \ldots, \beta_{80,IV})^\intercal$. The idea for the first choice was to take equidistant effect sizes in the range from 0.1 to 3 with alternating signs, i.e.

$$\boldsymbol{\beta}_I = (0.1,\ -0.2,\ 0.3,\ -0.4, \ldots,\ 2.9,\ -3,\ 0, \ldots,\ 0)^\intercal.$$

For $\boldsymbol{\beta}_{II}$, the exponential of the values 0.1, 0.2, 0.3, ... is taken and then again the effects get alternating signs for increasing absolute values, so that

$$\boldsymbol{\beta}_{II} = (exp(0.1),\ -exp(-0.2),\ exp(0.3), \ldots,\ exp(2.9),\ -exp(3),\ 0, \ldots,\ 0)^\intercal.$$

The third variation of the effect vector $\boldsymbol{\beta}$ consists of constant effect sizes that again have alternating signs. Accordingly,

$$\boldsymbol{\beta}_{III} = (c,\ -c,\ c,\ -c, \ldots,\ 0, \ldots,\ 0)^\intercal,$$

where c is set to 0.9 for uncorrelated covariates, i.e. for $\rho = 0$, and to 0.3 for correlated covariates, i.e. for $\rho = 0.5$.

The first 30 entries of $\boldsymbol{\beta}_{IV}$ were set to an equidistance sequence from 0.2 to 3 with an increment of 0.2. and every value occurring twice, once with a positive and once with a negative sign:

$$\boldsymbol{\beta}_{IV} = (0.2,\ -0.2,\ 0.4,\ -0.4, \ldots,\ 2.8,\ -2.8,\ 3,\ -3,\ 0, \ldots,\ 0)^\intercal.$$

In this case the effects were chosen to be not linear effects, but breakpoint effects, meaning that they are only unequal to zero whenever the values of the respective covariate are above a breakpoint. Since the covariates are all normally distributed with mean $\mu = 0$, the breakpoint was set at zero. In that case, we can write that

$$P(Y = 1|X = \boldsymbol{x}) = \frac{e^{\boldsymbol{x}^\intercal \boldsymbol{\beta}^*_{IV}}}{1 + e^{\boldsymbol{x}^\intercal \boldsymbol{\beta}^*_{IV}}}\ ,$$

with $\beta^{*(i)}_{j,IV} = \beta_{j,IV}\mathbb{I}(x^{(i)}_j > 0)$, where $\beta^{*(i)}_{j,IV}$ denotes the j-th entry of $\boldsymbol{\beta}^*_{IV}$ for the i-th observation, $x^{(i)}_j$ denotes the value of the j-th covariate for the i-th observation and $\mathbb{I}(\cdot)$ denotes the indicator function.

The values of the $\boldsymbol{\beta}$ vectors were chosen under the precondition that for all simulation settings the performance of a random forest measured by the AUC should be sufficiently good, more precisely, that the AUC should be at least 0.75 or higher. The AUC, short for *area under the curve*, is a performance measure for data with a binary response. For a sample of size n with predicted responses $\hat{y}_k$ for an observation k, $k \in \{1, \ldots, n\}$, $S(\hat{y}_i, \hat{y}_j)$ is summed up for all possible pairs of observations (i, j) with $y_i = 1$ and $y_j = 0$. $S(a, b)$ is either 1, if $a > b$, i.e. in the present case if the predicted responses in the pair are the same as the true responses, or 0.5 if $a = b$, or zero if $a < b$, i.e. if the predicted responses are reversed with respect to the true response values. The sum of $S(\hat{y}_i, \hat{y}_j)$ over all such pairs is divided by the total number of such pairs, which is $N^{(1)}$ times $N^{(0)}$, with $N^{(1)}$ and $N^{(0)}$ being the number of observations in the sample with response 1 and 0, respectively. (cf. Hanley and McNeil, 1982)

Generally, when the AUC is used to evaluate the performance of a classifier, it can be interpreted as the probability that the classifier ranks a randomly chosen pair of observations (i, j) with $y_i = 1$, $y_j = 0$ correctly. Technically, the AUC can take any value between 0 and 1, but a value of below 0.5 would mean that the classifier performs worse than random guessing and taking the exact opposite of what the classifier predicts would bring better results, so any reasonable classifier should have an AUC of more than 0.5. (cf. Fawcett, 2006)

To ascertain that this condition is fulfilled, the performance of a random forest carried out by the function `cforest()` from the `R` package `party` was evaluated by tenfold cross-validation, averaged over 100 simulated data sets per setting in the case of $n = 200$ to get sufficiently stable results. In the case of $n = 1000$, results of a tenfold cross-validation were averaged over twenty simulated data sets per settings, as there is less variation in the AUC for larger sample sizes and the computation time is considerably greater. The averaged results of this evaluation can be seen in Table 4 in Appendix A.1; they all meet the condition.

For $\boldsymbol{\beta}_{III}$, different values c were selected for the correlated and the uncorrelated setting because the performance of the forests built from data with no correlation between covariates is rather weak for a lower c, while for a higher c the permutation variable importance measure discriminates nearly perfectly between the influential and non-influential covariates in some settings, so that a comparison with the new VIM is hardly possible (cf. Section 5.2). Accordingly,

c was set to 0.9 for the simulated data with $\rho = 0$ to ensure a sufficiently good performance, and to 0.3 for the correlated case to mitigate the problem of the almost perfectly discriminating permutation importance.

In addition to the performance, the condition that the data should be fairly balanced was also taken into account when choosing the values for the effect vectors. For the linear effects, the exact values for the effects are of no consequence. Since the covariates are all normally distributed with mean $\mu = 0$, the linear predictor $\boldsymbol{x}^\intercal \boldsymbol{\beta}$ is also normally distributed with mean 0, independent of the values of $\boldsymbol{\beta}$. In consequence, a random variable $Z = \frac{e^{\boldsymbol{x}^\intercal \boldsymbol{\beta}}}{1+e^{\boldsymbol{x}^\intercal \boldsymbol{\beta}}}$ is logit-normally distributed, and more precisely, with $E(\boldsymbol{x}^\intercal \boldsymbol{\beta}) = 0$, Z is symmetrically distributed on the interval $[0, 1]$, so that $E(Z) = 0.5$ and therefore the response values of the simulated data will be on average perfectly balanced (cf. Frederic and Lad, 2003; Wutzler, 2021).

In the case of the breakpoint effects, it is more complicated, since $\boldsymbol{x}^\intercal \boldsymbol{\beta}^*_{IV}$ is not normally distributed. Intuitively, setting the values of $\boldsymbol{\beta}_{IV}$ in such a way that there are exactly the same effect sizes for the negative and the positive effects will lead to more balanced data than effect sizes like the ones in $\boldsymbol{\beta}_I$, where the absolute values of the negative effects are slightly higher than those of the positive effects. To assess whether data generated with breakpoint effects as described above does indeed lead to balanced data sets, the average proportion of the response values in 10000 data sets were observed for each combination of $\rho = 0$ or $\rho = 0.5$ and sample sizes of $n_1 = 200$ or $n_2 = 1000$, respectively. The results can be seen in Table 5 in Appendix A.2 and show that data generated in that way seem to be indeed well balanced on average.

4.2 Fixed and variating parameters

All random forests in this simulation study are generated with the function `cforest()` from the package `party` (Hothorn et al., 2021). Most of the hyperparameters are kept fixed, only the parameters `mtry`, which defines the number of covariates considered for splitting, and `maxdepth`, which defines the depth to which the trees are maximally grown by specifying the number of maximal layers, are variated in the simulation settings.

Most of the fixed parameters are set to their default values. For the parameter `teststat`, which defines the type of test statistic used, the default is "max", meaning that the maximum of the absolute values of the standardized linear statistic is used (opposed to "quad", i.e. using a quadratic form). The parameter `testtype` determines whether the value of the test statistic (which is the default) or the p-value (and what kind of adjustment) is used for the selection of the split variable. The parameter `mincriterion` defines, dependent on the setting of `testtype`, either the value of the test statistic or 1 - p-value that must be exceeded for a split. Its default (for `cforest()`; note that the default settings for `ctree()` are different) is set to qnorm(0.9), which corresponds to a significance level α of 0.1 (cf. Section 2.2; for further details, cf. Hothorn et al., 2006, and Hothorn et al., 2021).

For the parameters `minsplit` and `minbucket`, the default values of 20 and 7, respectively, are maintained in all simulation settings. Both of these parameters determine the tree depth by defining how much observations at least have to be in a node to be considered for splitting and the minimum number of observations in a terminal node, respectively. The number of trees per forest, determined by the parameter `ntree`, is set to 1000 in all simulation settings, which is twice as high as the default value of 500, as a higher number of trees ensures more stable variance importance estimates (cf. Genuer et al., 2008). The parameter `replace` was set to `TRUE`, its default, which means that bootstrap sampling was used for constructing the trees.

The default value of `mtry` is independently from the number of covariates 5, but for the simulation study, two other values are chosen for this parameter: firstly, the value is set to $0.257 \cdot p$, which rounded amounts to 21 for $p = 80$ covariates. This value is based on Probst et al., (2018), where $0.257 \cdot p$ was the optimal value according to hyperparameter tuning evaluated with the AUC. The second choice of `mtry` is the value which has generally been recommended as default for classification (cf. Section 2.1) and is indeed the default value for classification in other R packages dealing with random forests, namely `randomForest` and `ranger`.

Next to `mincriterion`, `minsplit` and `minbucket`, the parameter `maxdepth` may also be employed as stopping criterion. It controls the maximal depth of the trees by specifying the amount of layers that the trees must not exceed. For the simulation study, `maxdepth` was set either to its default value 0, which means that no restrictions are applied, or to 3, meaning that the trees must contain no more than three layers of splits. In that way, both forests with full trees and forests with rather small trees are considered in the simulation study.

4.3 Simulation structure

In summary, there are four choices of effects (defined by the four different $\boldsymbol{\beta}$-vectors), two covariance matrices, two variations of sample sizes, two different values for the parameter `mtry` and two different tree depths controlled by the parameter `maxdepth`, summing up to 64 different simulation settings. Each setting was repeated $n_{sim} = 100$ times, and for each iteration, variable importances were computed using both the new method by Hapfelmeier et al. (2014), and the standard permutation VIM. Both methods are implemented in `party`, controlled by the parameter `pre1.0_0` in the function `varimp()`.

			mtry = 21		mtry = 8	
effects	corr.	n	full trees	3 layers	full trees	3 layers
$\boldsymbol{\beta}_I$	$\rho = 0$	200	scenario 1	scenario 2	scenario 3	scenario 4
$\boldsymbol{\beta}_I$	$\rho = 0$	1000	scenario 5	scenario 6	scenario 7	scenario 8
$\boldsymbol{\beta}_I$	$\rho = 0.5$	200	scenario 9	scenario 10	scenario 11	scenario 12
$\boldsymbol{\beta}_I$	$\rho = 0.5$	1000	scenario 13	scenario 14	scenario 15	scenario 16
$\boldsymbol{\beta}_{II}$	$\rho = 0$	200	scenario 17	scenario 18	scenario 19	scenario 20
$\boldsymbol{\beta}_{II}$	$\rho = 0$	1000	scenario 21	scenario 22	scenario 23	scenario 24
$\boldsymbol{\beta}_{II}$	$\rho = 0.5$	200	scenario 25	scenario 26	scenario 27	scenario 28
$\boldsymbol{\beta}_{II}$	$\rho = 0.5$	1000	scenario 29	scenario 30	scenario 31	scenario 32
$\boldsymbol{\beta}_{III}$	$\rho = 0$	200	scenario 33	scenario 34	scenario 35	scenario 36
$\boldsymbol{\beta}_{III}$	$\rho = 0$	1000	scenario 37	scenario 38	scenario 39	scenario 40
$\boldsymbol{\beta}_{III}$	$\rho = 0.5$	200	scenario 41	scenario 42	scenario 43	scenario 44
$\boldsymbol{\beta}_{III}$	$\rho = 0.5$	1000	scenario 45	scenario 46	scenario 47	scenario 48
$\boldsymbol{\beta}_{IV}$	$\rho = 0$	200	scenario 49	scenario 50	scenario 51	scenario 52
$\boldsymbol{\beta}_{IV}$	$\rho = 0$	1000	scenario 53	scenario 54	scenario 55	scenario 56
$\boldsymbol{\beta}_{IV}$	$\rho = 0.5$	200	scenario 57	scenario 58	scenario 59	scenario 60
$\boldsymbol{\beta}_{IV}$	$\rho = 0.5$	1000	scenario 61	scenario 62	scenario 63	scenario 64

Table 1: Overview of simulation scenarios. "full trees" refers here and in the following to trees that are not restricted by a maximal number of layers, i.e. $maxdepth = 0$; "3 layers" refers to trees with $maxdepth = 3$.

Table 1 shows a overview of all 64 scenarios. They are sorted by effect vectors, and within the groups of effects by correlation structure, and then by sample size. The order of the numbering is not completely adhered to in Section 5 as the results for the scenarios with effects $\boldsymbol{\beta}_{IV}$ are discussed before those with $\boldsymbol{\beta}_{III}$. This is done because the variable importance measures are evaluated with correlation coefficients in scenarios with effects $\boldsymbol{\beta}_{IV}$, in the same way as with effects $\boldsymbol{\beta}_I$ and $\boldsymbol{\beta}_{II}$, while the importance measures are evaluated differently for scenarios with constant effects, as described in the following section.

4.4 Evaluation of results

Variable importance measures rank covariates according to their association to the response. Accordingly, the importance score of a covariate should be the higher the larger the effect size of this covariate is. To evaluate and compare both variable importance measures, correlations between the effect sizes and the variable importance values were calculated in the cases of variating effect sizes, i.e. for all simulation settings with $\boldsymbol{\beta}_I$, $\boldsymbol{\beta}_{II}$, and $\boldsymbol{\beta}_{IV}$.

The variable importance scores cannot be assumed to be normally distributed, as their distribution is much too right-skewed. That can be seen in Figure 1, where boxplots of the variable importance scores of all 80 covariates from the first five iterations of three different simulation settings are shown, one for each $\boldsymbol{\beta}$ vector with variating effect sizes (scenario 1, with effects $\boldsymbol{\beta}_I$; scenario 17, with effects $\boldsymbol{\beta}_{II}$; scenario 49, eith effects $\boldsymbol{\beta}_{IV}$; all three scenarios have $n = 200$, $\rho = 0$, $mtry = 21$ and $maxdepth = 0$; cf. Table 1 in Section 4.3). For that reason, Spearman's rank correlation coefficient was used, since it is a nonparametric measure that does not require normally distributed variables.

In the case of $\boldsymbol{\beta}_{III}$, all effects have the same size, which means that the influential covariates should all have similar variable importance scores that should be distinctly higher than the scores of the noise variables, which should be approximately zero. In order to evaluate the ability of a VIM to differentiate between influential and irrelevant covariates, the AUC was computed as it is done in Janitza et al. (2013), meaning that for each pair of an influential and an irrelevant covariate, the corresponding VI values were evaluated with $S(\cdot)$ as described in Section 4.1. The AUC can be interpreted here as the probability that a randomly drawn influential covariate has a higher variable importance than a randomly drawn noise variable. Possible AUC values range from 0.5 to 1, with 0.5 meaning that the variable importance measure is not able to discriminate between relevant and irrelevant covariates better than one would by random guessing. A value of 1 on the other hand indicates a perfect discrimination, i.e. the variable importance score of each influential covariate is higher than that of any noise variable. (cf. Janitza et al., 2013) For computing the AUC, the function `AUC()` from the R package `MLmetrics` (Yan, 2016) was used.

Correlation coefficients or AUCs were computed for each iteration of each simulation setting both for the new PropRandom importance and for the permutation importance, resulting in two times 100 values per setting. In all cases, the correlations/AUCs for both variable importances were very similar. To assess whether the small differences are statistically significant, Wilcoxon signed-rank tests were conducted on the pairs of correlations/AUCs for all simulation settings.

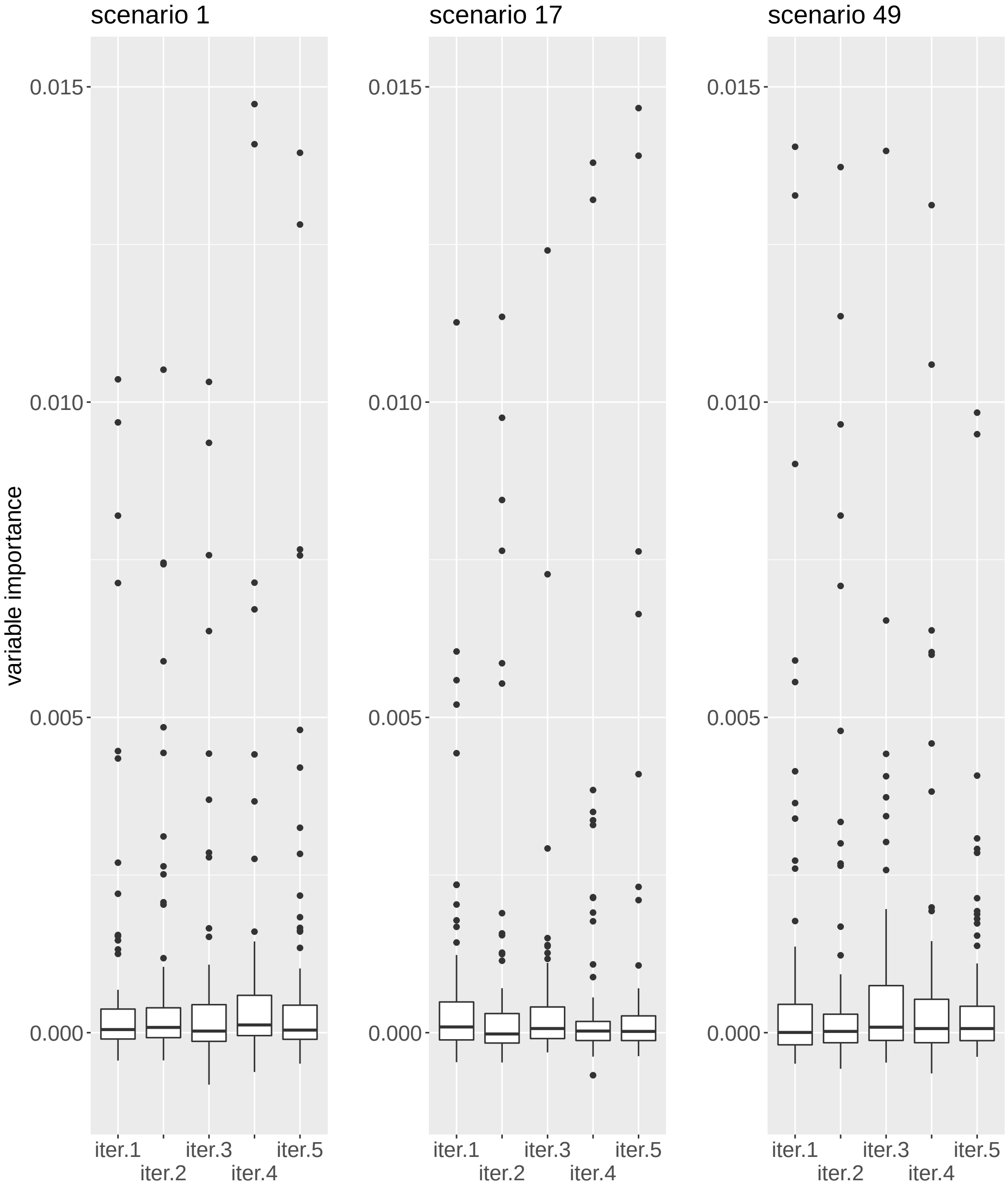


Figure 1: Variable importance values (measured with PropRandom) for all 80 covariates, shown for the first five iterations of three scenarios. For clarity, outliers above 0.015 were omitted, which concerns 2 values for scenario 1, 11 values for scenario 17 and one value for scenario 49.

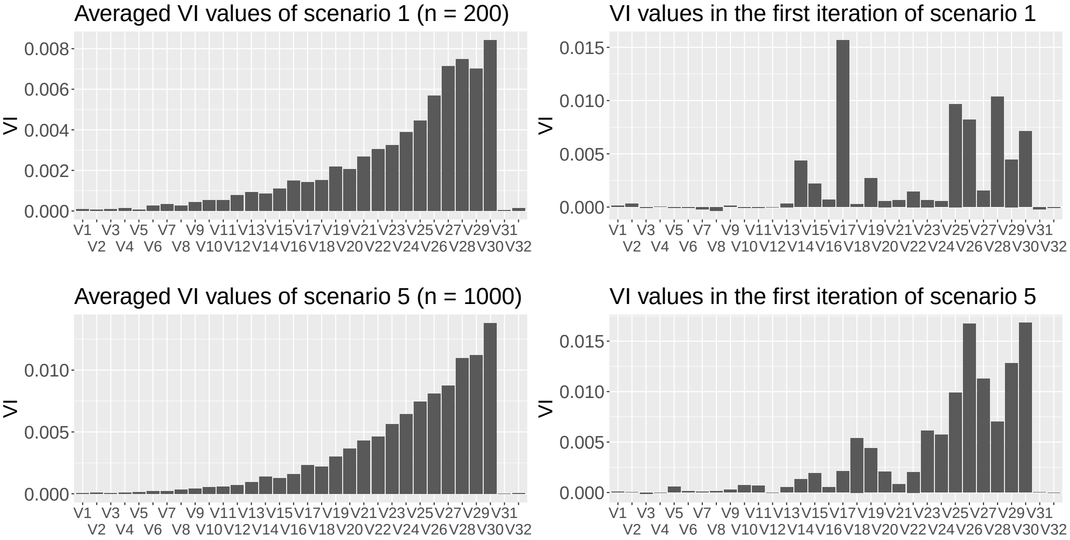
The Wilcoxon signed-rank test is a non-parametrical test that uses the ranks of observations instead of the raw values. It can be applied to one sample, or to two samples of paired data, and basically tests if the null hypothesis can be rejected that the median of the one sample, or the median difference in case of two matched samples, is zero. (cf. Hollander and Wolfe, 1973) In its functionality it serves the same purpose as a (paired) Student's t-test, but it does not assume a normal distribution. Since the correlation and AUC values cannot be assumed to be normally distributed in all settings (cf. Figures 4 - 11; at least some of the boxplots show considerable skewness), the Wilcoxon signed-rank test was preferred here to the t-test.

5 Results

5.1 Comparison of the correlations between effect sizes and variable importance values

Figures 4 - 9 show the correlations between the effect sizes and the variable importance scores for all scenarios with variating effect sizes. For each scenario each boxplot shows 100 correlations for the 100 iterations; the blue boxplots contain the correlations between the PropRandom importance values and the effect sizes, and the pink boxplots show the correlations between the permutation importance values and the effect sizes. Each figure displays either eight scenarios with no correlation between covariates ($\rho = 0$) or eight scenarios with blockwise correlation between covariates ($\rho = 0.5$). The plot in the first row of each figure shows the correlations for scenarios with a sample size of $n_1 = 200$, and the second row shows the correlation values for scenarios with sample size $n_2 = 1000$. Note that the scales of correlation values in each figure differs considerably between the two rows, as the different sample sizes affect the correlation strength.

Both the influence of the sample size and the variability of variable importance values for the simulated data sets is exemplified in Figure 2 by scenario 1 and 5, which have, apart from the sample size, exactly the same parameter settings ($\rho = 0$, $mtry = 21$, $maxdepth = 0$). The effect sizes are in both scenarios the absolute values of $\boldsymbol{\beta}_I$, which are monotonously and equidistantly increasing for the first 30 covariates. For the sake of clarity, only the 32 first covariates are considered in the plots, i.e. all influential covariates plus two noise variables. The left plot shows the PropRandom variable importance values averaged over all 100 iterations, and in the right plot the variable importances of only the first iteration are depicted. While the averaged variable importance values are, corresponding to the effect sizes, for the most part monotonously increasing over the first 30 variables, the variable importance values of only one simulation iteration do not reflect the pattern of the increasing effect sizes very well. Obviously, the variable importance measures on average provide a sensible variable ranking while for a single data set the importance values may deviate notably from the pattern of the effect sizes. It is not surprising that this deviation is more pronounced for small-sized data sets ($n = 200$) than for data sets of moderate size ($n = 1000$), since a larger amount of data can be expected to provide more stable variable importance estimates.



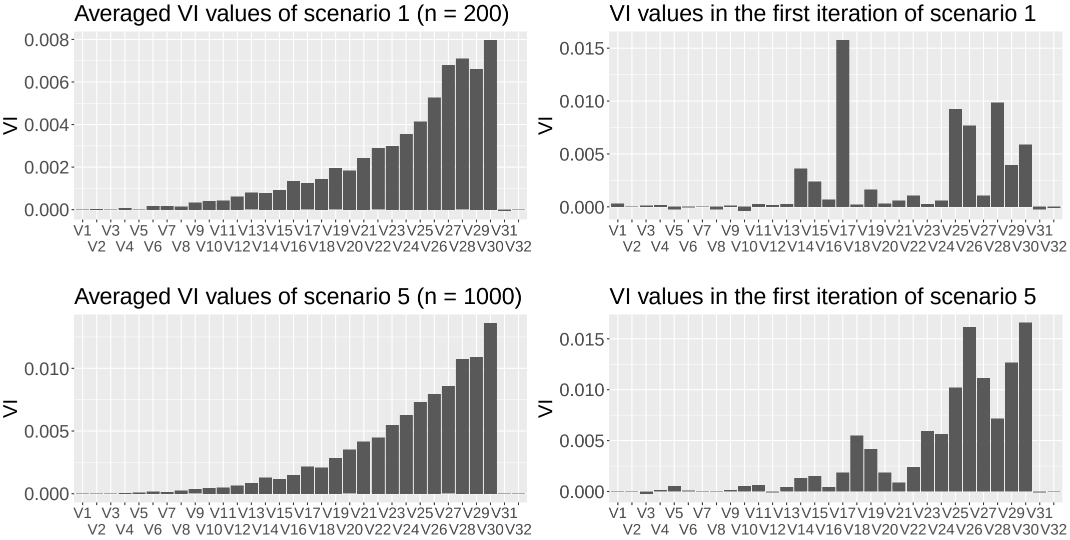
Averaged VI values of scenario 1 (n = 200)

VI values in the first iteration of scenario 1

Averaged VI values of scenario 5 (n = 1000)

VI values in the first iteration of scenario 5

Figure 2: VI values (PropRandom) of the first 32 covariates, for scenario 1 and scenario 5. On the left, the values are averaged over all 100 iterations; on the right, only the values of the first iteration are shown.



Averaged VI values of scenario 1 (n = 200)

VI values in the first iteration of scenario 1

Averaged VI values of scenario 5 (n = 1000)

VI values in the first iteration of scenario 5

Figure 3: VI values (permutation importance) of the first 32 covariates, for scenario 1 and scenario 5. On the left, the values are averaged over all 100 iterations; on the right, only the values of the first iteration are shown.

When comparing Figure 2 and 3, which show the PropRandom and the permutation importance values, respectively, for exactly the same settings, one can see that both measures lead in these scenarios to extremely similar rankings, both on average and for the single simulation iteration. In fact, for the averaged values the ranking is exactly the same for all influential covariates in the case of $n = 1000$ (scenario 5), and for the 23 highest ranked covariates in the case of $n = 200$ (scenario 1), and even the differences between the importance values of the individual covariates seem to be almost identical. For the example of the single iterations, the rankings are also very similar, and though there are some differences in the order of the ranking, the pattern of importance values looks extremely similar for both methods.

In Figure 4 and 5 the correlations between the effect sizes and the variable importance scores for the first sixteen scenarios can be seen, which means for the simulation settings with the effect sizes of $\boldsymbol{\beta}_I$; in Figure 4 for simulated data without correlation between covariates ($\rho = 0$), and in Figure 5 for simulated data with blockwise correlation between covariates ($\rho = 0.5$) as described in section 4.1. There are only small differences visible between the correlation values for the PropRandom importance (shown in blue) and the correlations for the permutation importance (shown in pink). For most of the eight scenarios in Figure 4, the median of the correlations is a bit higher for the PropRandom importance, only for scenario 3 (n = 200, mtry = 8, full trees) the medians are approximately the same, and for scenario 4 (n = 200, mtry = 8, maxdepth = 3) the median is a bit higher for the permutation importance. Both quartiles, represented by the lower and upper edges of a boxplots, are in six of the eight scenarios higher for the PropRandom importance. Only for the two scenarios with a small sample size and tree sizes restricted to three layers, the upper quartile is higher for the permutation importance. In regard to the variance of the correlation values, there seems to be only very little difference between the box sizes when comparing the PropRandom boxplots to the permutation boxplots.

For the four scenarios with $n = 1000$, the differences between the medians and the location of the boxes appear to be more pronounced than for the scenarios with a small sample size. This becomes also apparent in the results of two-sided paired Wilcoxon tests: there are no significant differences between the correlations for the first four scenarios with $n = 200$. In contrast to that, for three of the moderate size scenarios, namely scenario 5, 6 and 8, the Wilcoxon test yields p-values lower than 0.05, and in the case of scenario 5 and 8 even lower than 0.01 and 0.001, respectively. From the four scenarios with $n = 1000$, scenario 7 with $mtry = 21$ and $maxdepth = 0$ is the only one without a significant difference between the two importance measures. The p-values from the Wilcoxon tests for all scenarios are listed in Tables 2 and 3 in Section 5.4.

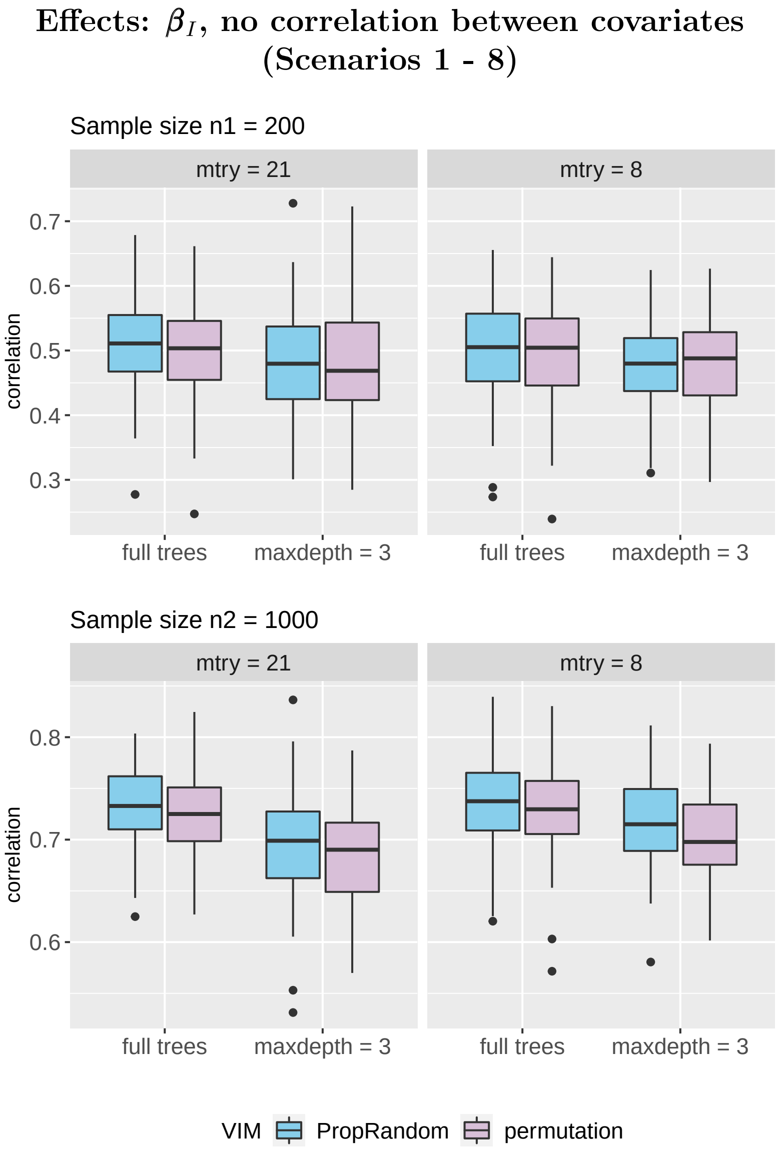


Figure 4: Correlation between effects $\boldsymbol{\beta}_I$ and VIs, for data with no correlation between covariates (scenarios 1 - 8). Each boxplot represents 100 correlation coefficients, computed in each iteration of the simulation both for PropRandom importance values (blue) and permutation importance values (pink).

Apart from the sample size, there is no clear pattern recognizable in these eight scenarios of how the different simulation settings affect the differences between the importance measures. Neither changes in mtry nor in maxdepth seem to influence the difference between PropRandom and the permutation importance much. What can be seen though is that three-layered trees for both importance measures lead to smaller correlation strengths than fully grown trees. It is not surprising that this influence of the tree depth is stronger for the larger sample sizes; because of the parameters minsplit and minbucket, which restrict the tree size depending on the size of nodes, most trees grown from a $n = 200$ data set will not have much more layers than three anyway.

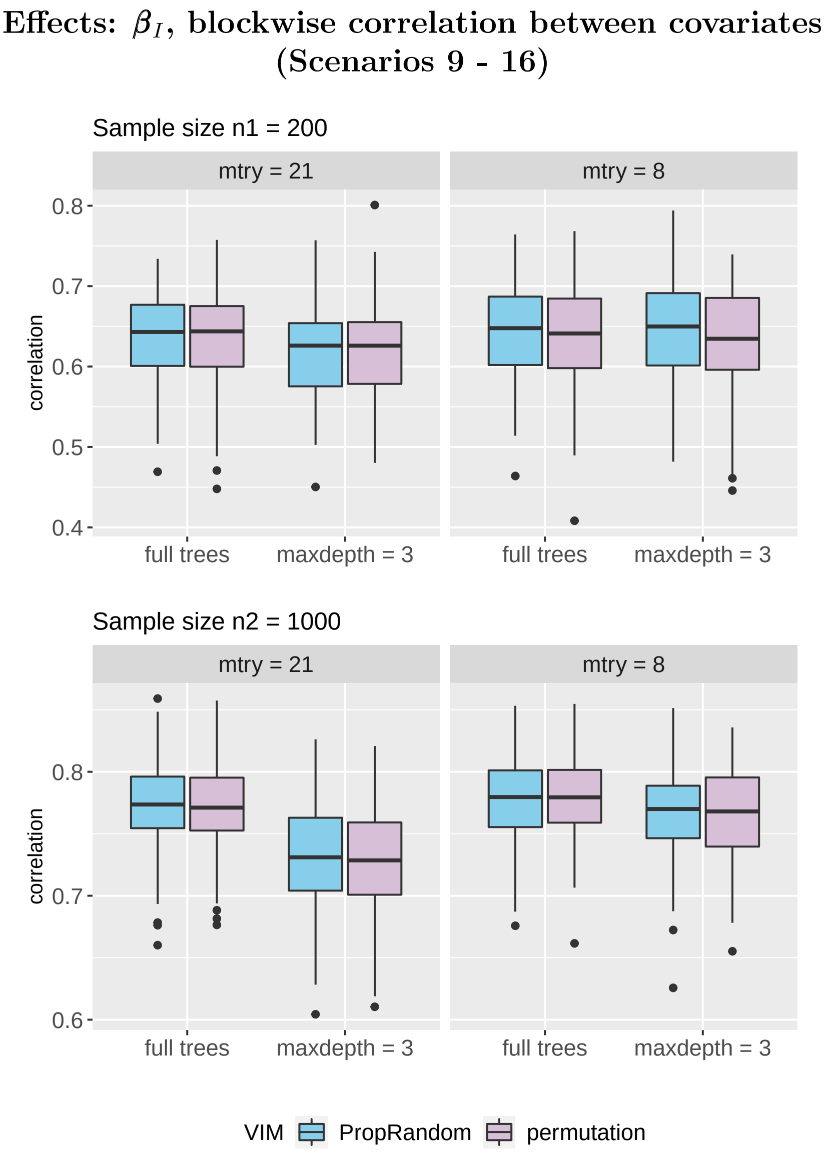


Figure 5: Correlation between effects β_I and VIs, for data with blockwise correlation between covariates (scenarios 9 - 16). Each boxplot represents 100 correlation coefficients, computed in each iteration of the simulation both for PropRandom importance values (blue) and permutation importance values (pink).

For scenarios 9 - 16, which are shown in Figure 5, there seem to be even less differences between the correlation boxplots for both measures than for the scenarios with $\rho = 0$. For scenarios 9, 10 and 15, the medians appear to be the same for both measures, while for scenario 13, 14 and 16 the median is minimally higher in the PropRandom boxplots. The largest differences, both with a greater median correlation for the PropRandom procedure, can be seen in scenarios 11 and 12, which represent simulation settings with $n = 200$ and $mtry = 8$. Among these two, the difference between boxplots is wider in scenario 12, which is also the only scenario of these eight in Figure 5 where the Wilcoxon test yields a significant result for an alpha level of 0.05, with a p-value of 0.04823. So here we have a significant difference only in one scenario with a sample size of 200, while for the first eight scenarios with no correlation between covariates only scenarios based on moderate-sized data sets showed significant differences between the variable importance measures.

What stands out a bit in Figure 5 is that in scenario 16 the boxplot of correlations for the permutation importance values has a visibly larger box, i.e. a larger interquartile range, than that of correlations for the PropRandom importance, while in the other scenarios each two boxes have approximately the same size. A similar effect but to a lesser extent could be seen in Figure 4, where the permutation boxes were visibly larger in scenarios 4 and 2, which have both, like scenario 16, settings with $maxdepth = 3$. The sample size, however is 1000 for scenario 16 and 200 for the other two scenarios, and besides in all three cases no significant differences between the correlations can be detected.

Figures 6 and 7 show the sixteen scenarios with the effects of $\boldsymbol{\beta}_{II}$, i.e. where the effect sizes are also monotonously increasing, but exponentially increasing and not linearly like it is the case for $\boldsymbol{\beta}_I$. Again, the differences between the correlations for the two importance measures are very small. In most pairs of correlation boxplots in Figure 6, the median is higher for the PropRandom importance, only in the last two scenarios, 23 and 24, with $n = 1000$ and $mtry = 8$, the median is a bit higher for the permutation importance. Regarding the results of the Wilcoxon signed-rank tests, scenario 18 ($n = 200$, $mtry = 21$), shows a significant difference between correlations with a p-value lower than 0.01, and for scenario 20 (with the same parameters as scenario 18, except that here $mtry = 8$) the test result is significant for $\alpha = 0.1$, while the p-values are well above 0.1 for the other six scenarios (cf. Table 2 in Section 5.4).

When comparing the interquartile ranges between the pairs of correlations, i.e. the sizes of the boxes, the most noticeable difference can be seen in scenario 18, where the range of the permutation correlations is greater than that of the PropRandom correlations. This is also the case for scenarios 19 and 22, but much less marked. For the other five scenarios, hardly any difference in the size of the boxes is discernible. In general, the variance of the correlation is always greater for the scenarios with $n = 200$ than for the scenarios with $n = 1000$ what is disguised a bit in Figures 4 to 11 by the fact that the plots for different sample sizes have different scales and larger ranges are covered in the plots with $n = 200$ than in the plots with sample sizes of 1000.

Effects: β_{II}, no correlation between covariates (Scenarios 17 - 24)

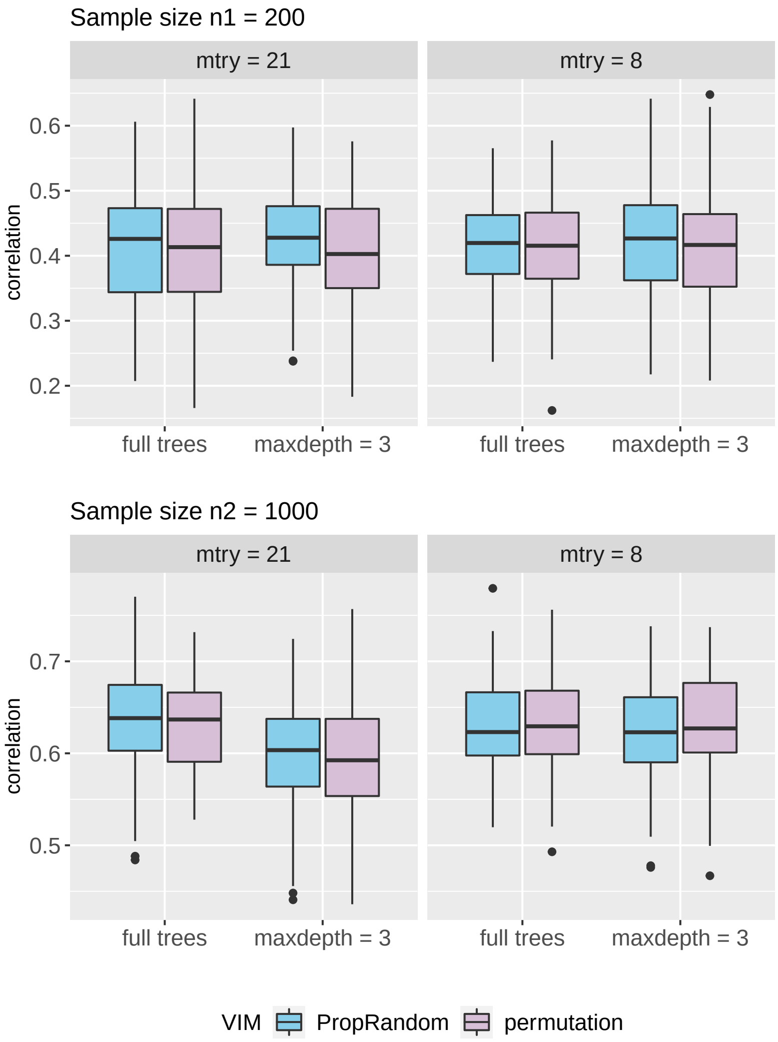


Figure 6: Correlation between effects β_{II} and VIs, for data with no correlation between covariates (scenarios 17 - 24). Each boxplot represents 100 correlation coefficients, computed in each iteration of the simulation both for PropRandom importance values (blue) and permutation importance values (pink).

The scenarios which are shown in Figure 7 have all simulation settings with the effects of β_{II} and blockwise correlation between covariates. For most of these scenarios, the medians of the PropRandom and the permutation correlations are very close together or even approximately equal. The widest differences can be seen in scenarios 26 and 30, which have both the same parameter settings ($mtry = 8$ and $maxdepth = 3$) except for the sample size. In both cases, the median correlation is higher for PropRandom than for the permutation importance, but the result of the Wilcoxon test is not significant for either of these scenarios.

Effects: β_{II}, blockwise correlation between covariates (Scenarios 25 - 32)

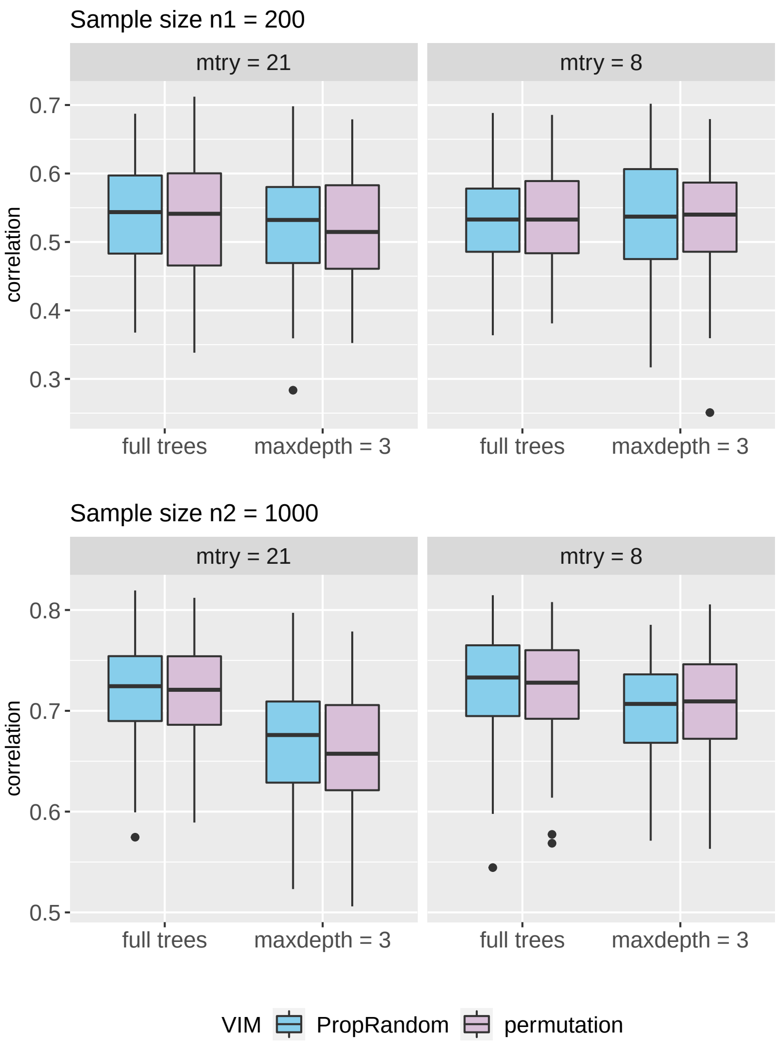


Figure 7: Correlation between effects $\boldsymbol{\beta}_{II}$ and VIs, for data with blockwise correlation between covariates (scenarios 25 - 32). Each boxplot represents 100 correlation coefficients, computed in each iteration of the simulation both for PropRandom importance values (blue) and permutation importance values (pink).

There are some visible differences in the box sizes for all scenarios with sample size 200, but while for the first three, namely scenario 25, 26 and 27 the box is a bit larger for the permutation importance, in scenario 28 the box is larger for the PropRandom importance, and this is also the widest difference among these four scenarios. For the scenarios with $n = 1000$, the box sizes are more even between the two variable importance measures.

There is also some variation regarding the box sizes between the pairs of boxplots, for both sample sizes, in scenarios 49 - 56 (shown in Figure 8), which are the scenarios with the breakpoint effects $\boldsymbol{\beta}^{*}_{IV}$ (cf. Section 4.1) and data with no correlation between covariates. The medians are almost the same for half of these eight scenarios. For scenarios 49 and 53, which both have the parameter settings $mtry = 21$ and $maxdepth = 0$, the median is higher for the

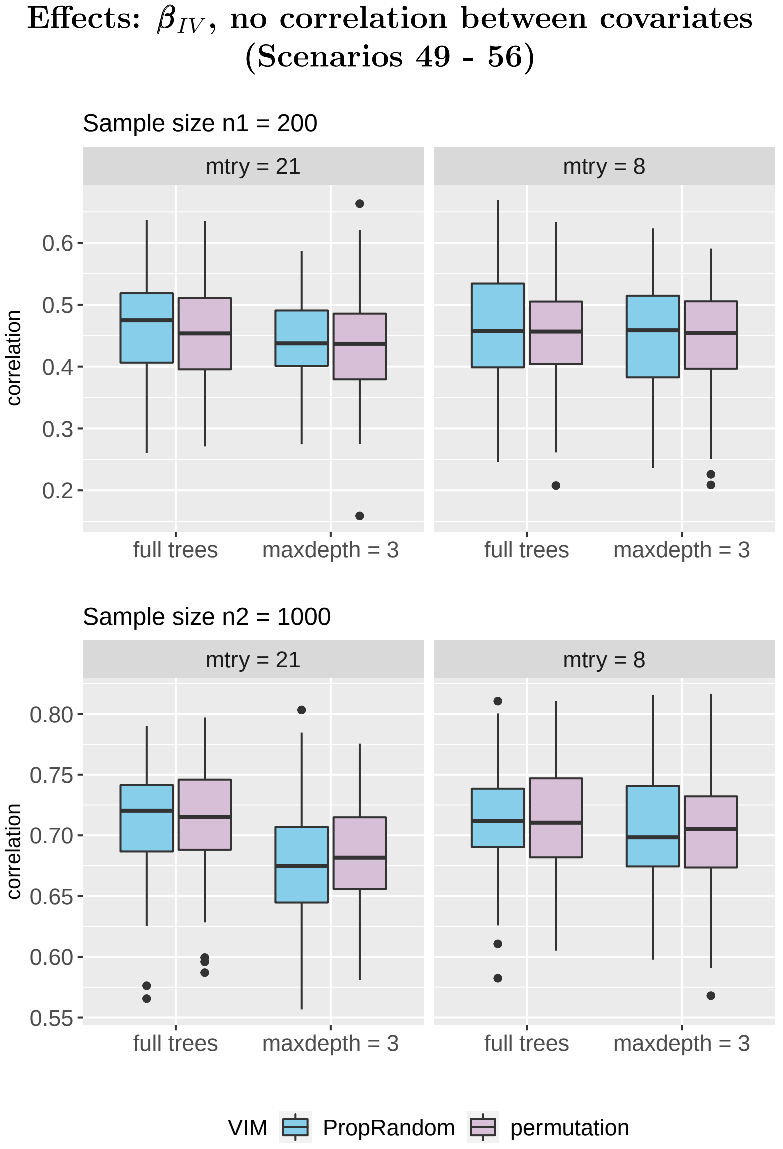


Figure 8: Correlation between effects $\boldsymbol{\beta}_{IV}$ and VIs, for data with no correlation between covariates (scenarios 49 - 56). Each boxplot represents 100 correlation coefficients, computed in each iteration of the simulation both for PropRandom importance values (blue) and permutation importance values (pink).

PropRandom correlations, while for scenarios 54 and 56, which both correspond to a sample size of 1000 and trees with maximally three layers, the median is higher for the permutation importance. In the case of scenario 54, also both quartiles are higher for the permutation importance, and it is the only scenario among the eight shown in this figure, where the Wilcoxon test yields a significant result, with a p-value of 0.035. Moreover, it is the only scenario among all 64 scenarios, where the permutation correlation is significantly higher than the PropRandom correlation.

Effects: β_{IV}, blockwise correlation between covariates (Scenarios 57 - 64)

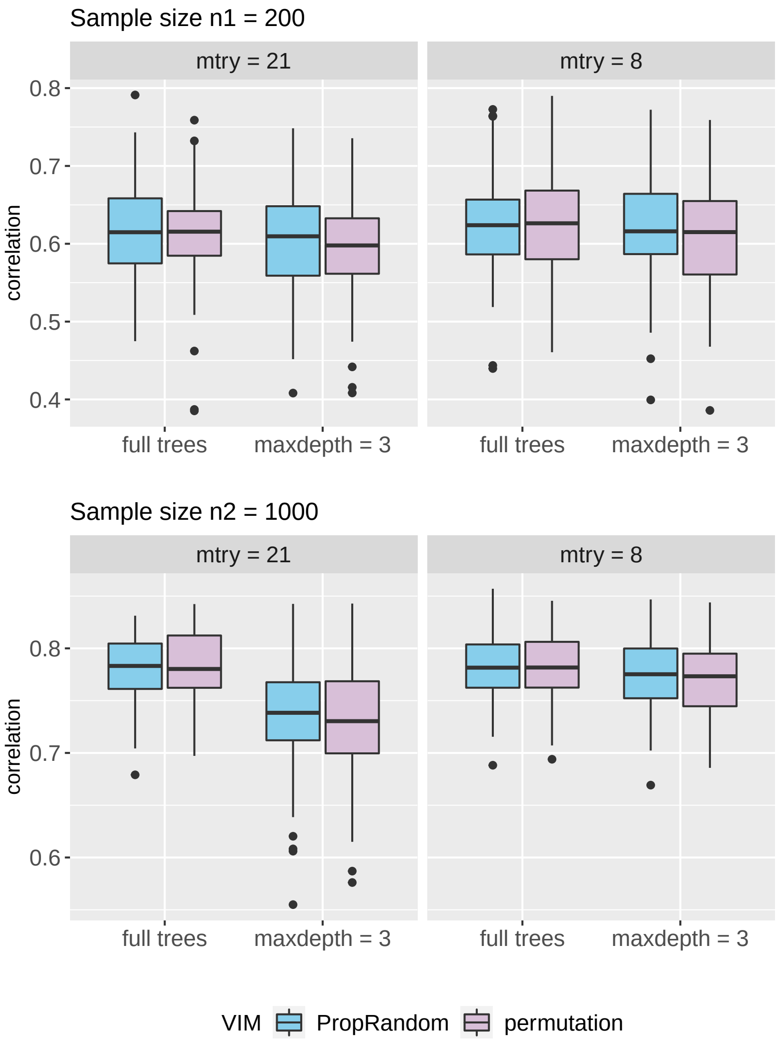


Figure 9: Correlation between effects β_{IV} and VIs, for data with blockwise correlation between covariates (scen. 57 - 64). Each boxplot represents 100 correlation coefficients, computed in each iteration of the simulation both for PropRandom importance values (blue) and permutation importance values (pink).

In Figure 9, scenarios with the same settings as in Figure 8 are shown, with the only difference that here the covariates in the simulated data sets are blockwise correlated with $\rho = 0.5$. The medians appear to be approximatively the same in the boxplot pairs of all scenarios except for scenarios 58 and 62, both with parameter settings $mtry = 21$ and $maxdepth = 3$, where the median is higher for the correlation between effect sizes and PropRandom variable importances. However, in both cases the median differences are not significant according to the results of the Wilcoxon test (cf. Table 3, Section 5.4).

The lowest p-value among these eight scenarios can actually be found in scenario 60 ($n = 200$, $mtry = 8$, $maxdepth = 3$), where the Wilcoxon test result is significant for $\alpha = 0.1$. As mentioned before, the medians are here almost the same for both importance measures. More precisely, the median is 0.616 when considering the correlation between effect sizes and PropRandom importance values, and 0.615 in the case of the permutation importance, so that the difference between the medians is about 0.001. Both quartiles, however, are higher for the PropRandom importance, and the median difference between both correlations, i.e. the median of all 100 differences between the correlation coefficients for both importance measures, is almost 0.01, which is nearly ten times higher than the difference between medians. This is a good example of how the difference between medians is not the same as the median difference, and while we can assess the former directly by looking at the boxplots in Figures 4 - 11, the latter is what is actually tested with the Wilcoxon test for being unequal to zero.

5.2 Comparison of the AUCs for constant effect sizes

Figures 10 and 11 show all scenarios with constant effect sizes c. The constant effect size c is 0.9 in scenarios 33 - 40, where $\rho = 0$ and 0.3 in scenarios 41 - 48, where $\rho = 0.5$ (cf. Section 4.1). The AUCs are relatively high for both importance measures in all scenarios. In the scenarios with a sample size of 1000, the median AUCs are close to 1 for data with $\rho = 0$ and equal to 1 for data with correlations between covariates (for both importance measures), which means that for scenarios 61 - 64 at least in half of the cases both variable importance measures manage to discriminate perfectly between relevant and irrelevant covariates.

When comparing the AUCs for the PropRandom importance and the permutation importance in Figure 10, one can see that the median AUC is for all eight scenarios either approximately the same or a little bit higher for the PropRandom importance. Scenario 38 has a much larger variance than the other scenarios with $n = 1000$. That makes it harder to visually assess differences between the other three boxplots, but by calculating the difference in medians, one can see that it is very small in all four scenarios. Among the small size scenarios, scenario 36, with $n = 200$, $mtry = 8$ and $maxdepth = 3$, has the most pronounced difference between AUCs, and the Wilcoxon test yields a p-value of 0.014. This is the only significant result among the eight scenarios in Figure 10.

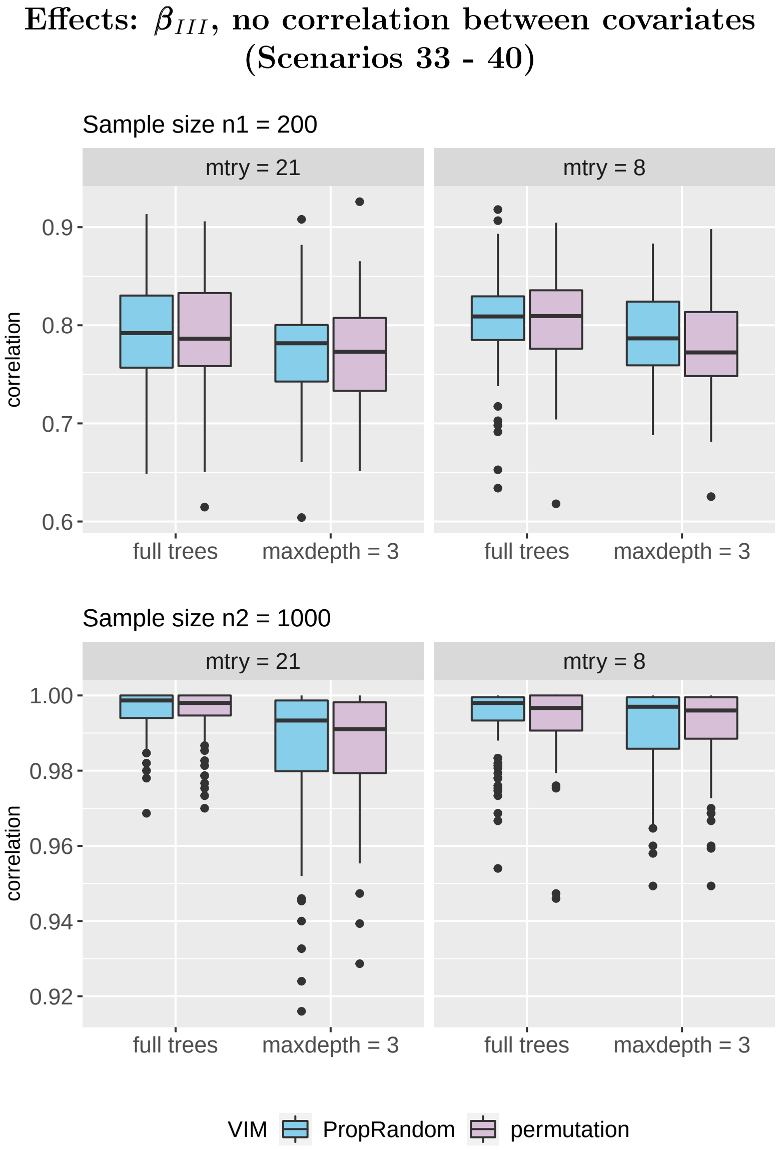


Figure 10: AUCs evaluating the discriminative ability of VIMs, for constant effects β_{III} and data with no correlation between covariates (scenarios 33 - 40). Each boxplot represents 100 AUCs, computed in each iteration of the simulation both for PropRandom importance values (blue) and permutation importance values (pink).

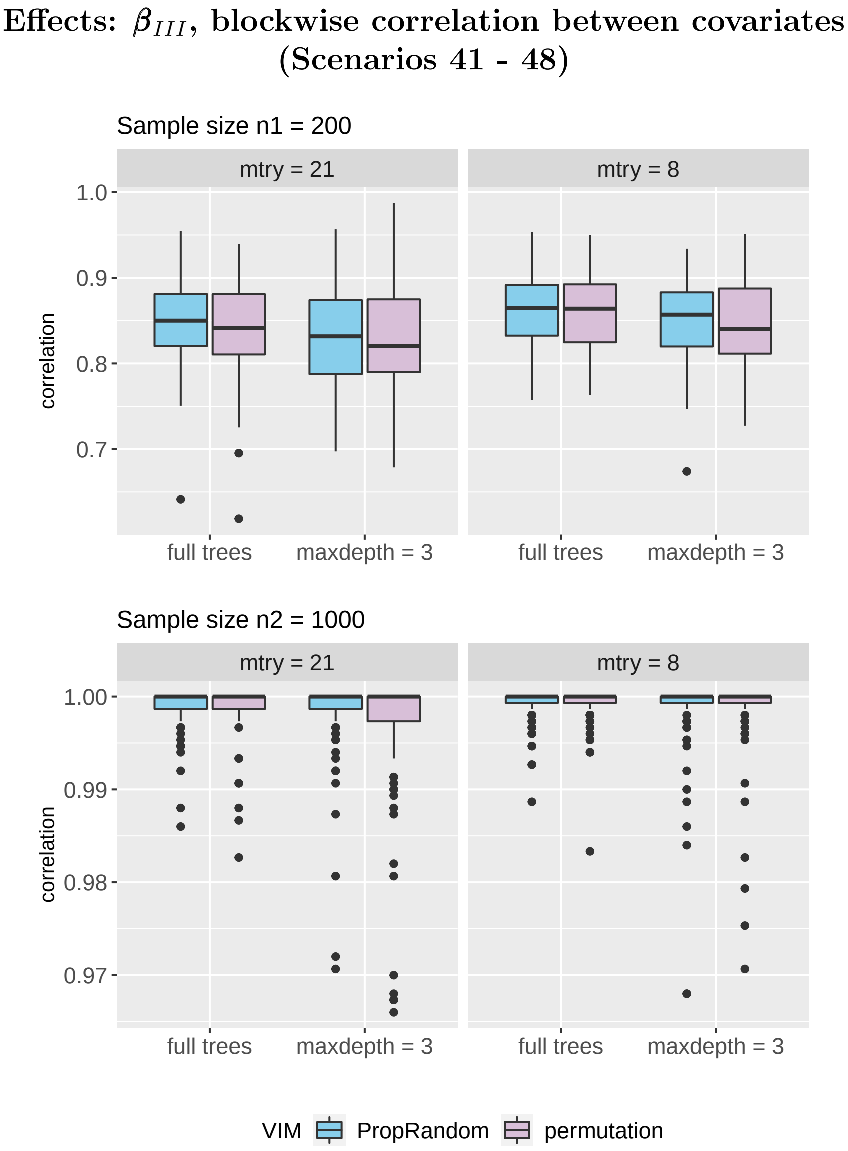


Figure 11: AUCs evaluating the discriminative ability of VIMs, for constant effects β_{III} and data with blockwise correlation between covariates (scenarios 41 - 48). Each boxplot represents 100 AUCs, computed in each iteration of the simulation both for PropRandom importance values (blue) and permutation importance values (pink). For clarity, one outlier of the PropRandom AUC in scenario 46 ($n = 1000$, $mtry = 21$, $maxdepth = 3$) with a value of 0.92333 was omitted.

Regarding the scenarios with $\rho = 0.5$ and a small sample size of 200, the medians are all higher for the PropRandom importance, though for scenario 43 the difference is so small that it can hardly be visually ascertained. Scenario 41 is the only scenario among these four where also the quartiles and both minimum and maximum are higher for the PropRandom importance. The difference between the AUCs is significant, at least for a significance level of 0.1. There is no significant difference between the AUCs for the other three scenarios.

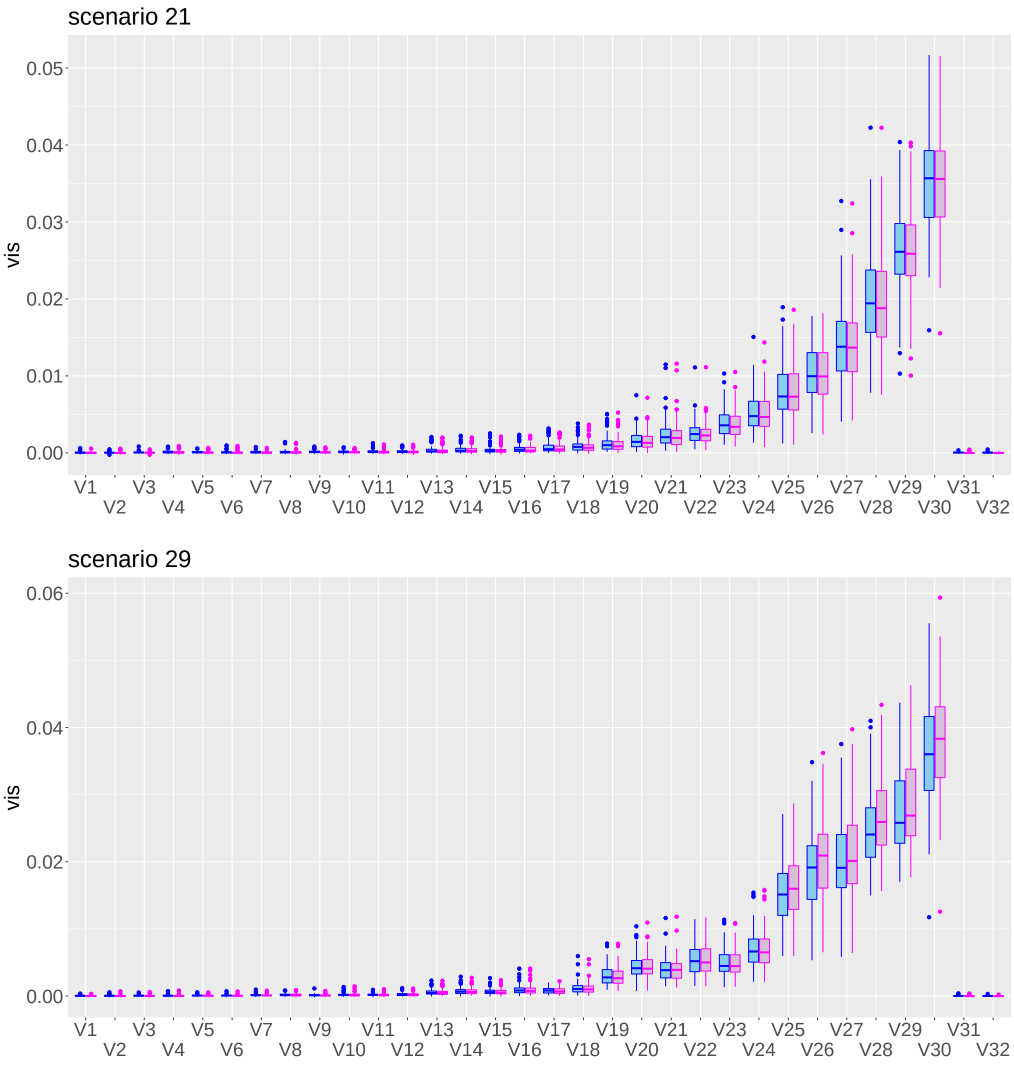
In the scenarios with correlation between covariates and a sample size $n = 1000$, the AUC boxplots are extremely similar. As mentioned before, the median is 1 for both importance measures in all four scenarios, and since 1 is the highest possible value for the AUC, this means that also the upper quartile and the maximum are the same for both importance measures in all scenarios. For scenarios 45, 47 and 48 the lower quartile and the lower whisker seem to be exactly the same for both importance measures, and computing the concrete values shows that they are indeed identical for the PropRandom importance and the permutation importance in these three scenarios. Only the outliers differ between the importance measures, but they are mostly also in a similar range.

In scenario 46 there is a visible difference: both the lower quartile and the end of the lower whisker are higher for the PropRandom importance. The differences are very small in absolute numbers, but in relation to the small range of the boxplots in scenarios 45 - 48, they are not negligible. Indeed, the difference between AUCs is significant in scenario 46, with a p-value of 0.0466. Just from the boxplots, it is not absolutely clear in this case what the direction of the difference is, especially since there is an extremely low outlier in the PropRandom AUCs that was omitted in Figure 11 for the sake of clarity. The median difference, computed for all 100 pairs of AUCs results in zero. Yet when all pairs are excluded where the difference of the AUCs is zero, which are 49 of the 100, and the median of the remaining 51 is taken (which is what `wilcox.test` in `R` actually does), this median is positive. Since the test was taken for the differences of the PropRandom AUCs minus the permutation AUCs, we can conclude that the discriminative ability, evaluated by means of the AUC, is significantly higher for the PropRandom importance in that scenario.

5.3 Comparison of the variable importance values

Not only the correlations between variable importance values and effect sizes are very similar between the two importance measures, but also the variable values themselves, as was already shown for the example of scenarios 1 and 5 in the beginning of Section 5.1. Another example for the closeness of the importance values from both measures can be seen in Figure 12. Since both variable importance measures compute the importance values in almost the same manner, more precisely, in both cases the importance values correspond to the differences between two OOB-accuracies, the absolute values of the two measures can be directly compared.

Figure 12 shows for the example of scenarios 21 and 29 the importance values of each variable computed by both measures, for all 100 iterations condensed in one boxplot, respectively. The scenarios have the same settings except for ρ, which is 0 in scenario 21 and 0.5 in scenario 29. The effects in these scenarios are those of $\boldsymbol{\beta}_{II}$, the sample size is 200, $mtry$ is 21 and the trees are not restricted in regard to their number of layers. As the noise variables all have very similar importance values, only the first two are included in the figure. The importance values of the PropRandom procedure are shown in blue and those of the permutation importance in pink.


scenario 21

scenario 29

Figure 12: PropRandom (blue) and permutation (pink) variable importance values for scenarios 21 and 29. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

One can see that though the importance measures apparently do not separate the smaller effect sizes very well from each other and from the noise variables, at least for the larger effect sizes the increasing structure is reflected quite well by the importance values, though the variable importances in scenario 29 are apparently influenced by the correlation structure as well. The blocks of interlocked correlation between two times three covariates, as described in Section 4.1, can be observed very clearly for the larger effect sizes, i.e. for covariates V19 to V24 and V25 to V30. The effect that both variable importance measures tend to attribute higher importance values to covariates that are correlated to more influential covariates has also been remarked upon in Hapfelmeier et al. (2014). Here, we can see for example how the importance of covariate V25 is markedly higher than that of V24 because of its correlation to covariates V27 and V29. The variable importances of covariates V20 and V22 are even a bit higher than those of covariates V21 and V23, respectively, because of their correlation to V24.

Something else can be seen when comparing the importance values for scenario 21 and scenario 29: in the former, the PropRandom importances tend to be a bit higher than the permutation importances, while in the latter it seems to be the other way round. When the paired importance values for all the other scenarios (cf. Figures 16 - 31 in Appendix A.3) are scrutinized under that aspect, one can see that in most cases the variable importance values seem to be on average higher for the PropRandom importance measure than for the permutation importance measure. Only in those scenarios which have a variating effect size, i.e. $\boldsymbol{\beta}_I$, $\boldsymbol{\beta}_{II}$ or $\boldsymbol{\beta}_{IV}$, correlation between covariates and a sample size of 1000, the permutation importance tends to attribute higher importance values than PropRandom does. Apparently, the different settings of *mtry* and *maxdepth* have no influence on this pattern, and neither correlation between covariates in combination with a small sample size nor a sample size of 1000 in combination with $\rho = 0$ leads to higher permutation importance values.

A higher importance value could be interpreted as a better ability to discriminate between relevant and irrelevant covariates of the respective importance measure. Looking at the importance values for the noise variables in detail, however, shows that the PropRandom importance also seems to attribute higher importance values to the noise variables than the permutation importance does. The question now is, in the cases where the PropRandom importance values are higher for the influential covariates than the corresponding permutation importance values, are they also further from the average PropRandom importance value of a noise variable than the corresponding permutation importance value is from the average permutation importance value of a noise variable? In order to assess whether this is the case, the mean importance value minus the mean importance value of all noise variables over all simulation iterations of the respective scenario were computed with both importance measures for each influential covariate. For these mean importance values that are scaled in relation to how far they are from the mean noise variable importance value, the differences were computed, by subtracting the permutation importance value from the PropRandom importance value, so that a positive result means that PropRandom discriminates better between relevant and irrelevant covariates and a negative result means that the permutation importance is better able to discriminate.

These differences are shown for four scenarios in Figure 13. The scenarios differ only in terms of sample size and correlation structure, and they confirm the pattern described above: in the three scenarios where either $n = 200$, or $\rho = 0$, or both, the PropRandom importance is, at least for the more influential covariates, on average higher than the permutation importance, even though the average noise variable importance value is subtracted out. Tendentially, the difference is higher for covariates with larger effect sizes, while the difference is near zero or has even a low negative value for covariates with a very weak effect. In scenario 29, however, where $n = 1000$ and $\rho = 0.5$, the difference is near zero for all covariates except the six most influential, and for them the difference is negative, i.e. the average permutation importance is higher.

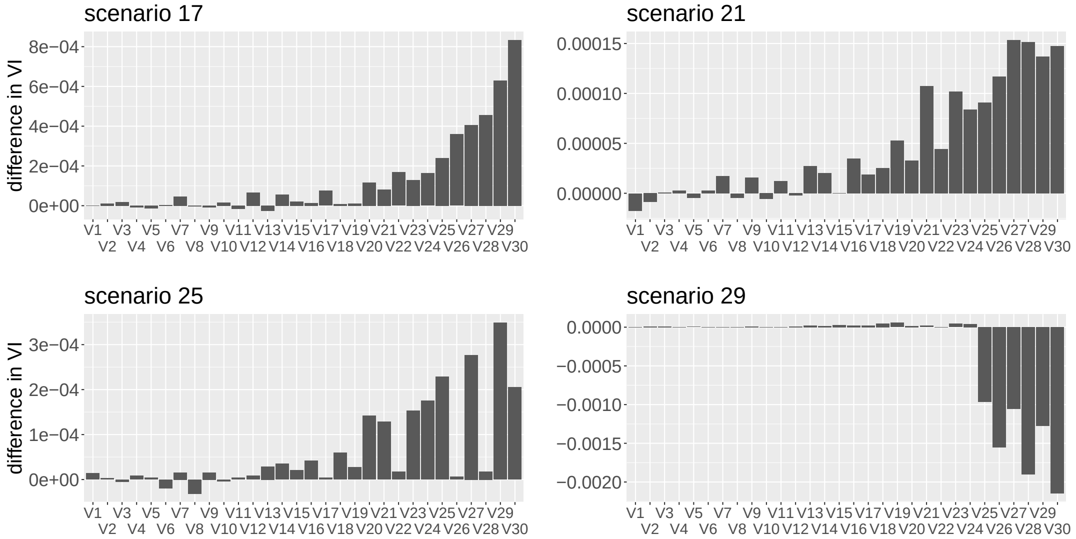


Figure 13: Differences between average PropRandom and permutation VIs for scenarios 17, 21, 25 and 29. All these scenarios have the effects $\boldsymbol{\beta}_{II}$ and the parameter settings $mtry = 21$ and $maxdepth = 0$ but differ in terms of the sample size and/or the correlation structure.

Figures 32 - 39 in Appendix A.4 show that the average permutation importance is not only in the case of $\boldsymbol{\beta}_{II}$ higher for the most influential covariates when the sample size is 1000 and the covariates are correlated, but that the same pattern can be seen for $\boldsymbol{\beta}_{I}$ and $\boldsymbol{\beta}_{IV}$. For $\boldsymbol{\beta}_{III}$, however, where all covariates have an equally strong influence, this effect can not be observed, in these scenarios the PropRandom importance tends to be higher than the permutation importance for all settings, regardless of the sample size or correlation structure. There is no obvious reason for that pattern, but apparently for a large enough sample size the permutation importance becomes more inflated for correlated and highly relevant covariates than the PropRandom importance.

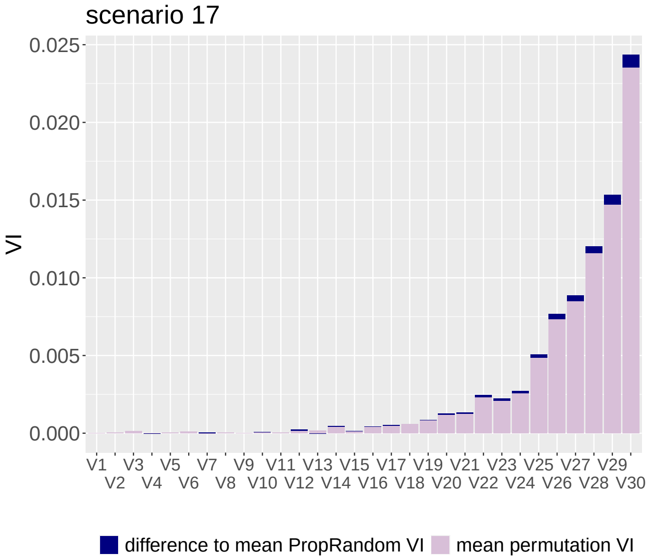


Figure 14: Mean permutation VIs for the influential covariates in scenario 17 are shown in pink. The differences between mean PropRandom VIs and permutation VIs are shown in dark blue.

Of course, it is important to consider that these differences are either way in all cases very small. The absolute values of these differences are all smaller than 0.001, most even considerably smaller. How small the differences actually are is illustrated in Figure 14 by the example of scenario 17. It shows the mean permutation importance values in pink and their difference to the mean PropRandom importance values in blue, so that for a positive difference the pink and blue parts of the bars sum up to the mean PropRandom VIs. For the weaker covariates, both the variable importances and the differences are so small that they can hardly be discerned in this figure.

5.4 Summary of results

The main result of the simulation study is that the two variable importance measures that were analysed perform very similar in all simulation settings. Not only the rankings of variables but even the variable importance values themselves are very close to each other for the two methods. There are small differences in the performance, evaluated by correlation in the case of variating effect sizes and by AUC in the case of constant effect sizes, between the measures, mostly in favour of the new PropRandom approach, but only very few of these differences could be shown to be significant. The p-values of the Wilcoxon tests for all scenarios are listed in Tables 2 and 3.

			mtry = 21		mtry = 8	
effects	corr.	n	full trees	3 layers	full trees	3 layers
β_I	$\rho = 0$	200	1 0.3096	2 0.8218	3 0.1833	4 0.8811
β_I	$\rho = 0$	1000	5 0.0089	6 0.0173	7 0.1197	8 0.0003
β_I	$\rho = 0.5$	200	9 0.9370	10 0.6217	11 0.2735	12 0.0482
β_I	$\rho = 0.5$	1000	13 0.4682	14 0.6611	15 0.9507	16 0.6536
β_{II}	$\rho = 0$	200	17 0.4608	18 0.0031	19 0.4734	20 0.0792
β_{II}	$\rho = 0$	1000	21 0.8514	22 0.6711	23 0.9630	24 0.1352
β_{II}	$\rho = 0.5$	200	25 0.8865	26 0.4993	27 0.4777	28 0.9739
β_{II}	$\rho = 0.5$	1000	29 0.7219	30 0.1028	31 0.5787	32 0.2631

Table 2: p-values for comparing the correlation between effect sizes and VIs measured by PropRandom and the permutation importance, respectively, with a two-sided Wilcoxon signed-rank test. The number above each p-value indicates the respective scenario. p-values lower than 0.05 are coloured in blue, p-values lower than 0.1 are coloured in dark green.

In summary, it can be said that for only eleven out of all 64 scenarios the p-value is lower than 0.1, of which only eight p-values are lower than 0.05. Only one of these p-values indicates a significantly higher performance of the permutation importance while in all other cases with significant p-values the correlation/AUC is higher for the PropRandom importance measure. The significantly higher performance of the permutation importance occurs in scenario 54, i.e. effects $= \boldsymbol{\beta}^*_{IV}$, no correlation between covariates, sample size of 1000, $mtry = 21$ and $maxdepth = 3$, but since it is the only significant difference in that direction, there can hardly be anything concluded from it.

			mtry = 21		mtry = 8	
effects	corr.	n	full trees	3 layers	full trees	3 layers
$\boldsymbol{\beta}_{III}$	$\rho = 0$	200	33 0.4483	34 0.4734	35 0.9819	36 0.0140
$\boldsymbol{\beta}_{III}$	$\rho = 0$	1000	37 0.2881	38 0.8547	39 0.2288	40 0.4413
$\boldsymbol{\beta}_{III}$	$\rho = 0.5$	200	41 0.0768	42 0.3967	43 0.3297	44 0.3993
$\boldsymbol{\beta}_{III}$	$\rho = 0.5$	1000	45 0.3531	46 0.0466	47 0.6887	48 0.7505
$\boldsymbol{\beta}_{IV}$	$\rho = 0$	200	49 0.1468	50 0.5103	51 0.1416	52 0.7478
$\boldsymbol{\beta}_{IV}$	$\rho = 0$	1000	53 0.9616	54 0.0354	55 0.8460	56 0.6799
$\boldsymbol{\beta}_{IV}$	$\rho = 0.5$	200	57 0.4311	58 0.4809	59 0.6938	60 0.0944
$\boldsymbol{\beta}_{IV}$	$\rho = 0.5$	1000	61 0.5348	62 0.2544	63 0.5191	64 0.4650

Table 3: p-values for comparing the correlation between effect sizes and VIs measured by PropRandom and the permutation importance, respectively, with a two-sided Wilcoxon signed-rank test. The number above each p-value indicates the respective scenario. p-values lower than 0.05 are coloured in blue, p-values lower than 0.1 are coloured in dark green.

As for the instances of a significantly higher performance of the PropRandom procedure, they are spread relatively evenly over the different settings. Four out of ten instances occur for the effects of $\boldsymbol{\beta}_I$, but for either of the other effects at least one significant result can be found as well. For *mtry*, the sample size and the correlation structure, the significant results occur almost equally often in both alternatives. For *maxdepth*, eight out of ten significant differences belong to a scenario with a restriction of tree layers, yet the significant difference in favour of the permutation importance also occurred in a setting with a restricted tree size. All in all, there is no clear pattern visible that either PropRandom or the permutation importance are distinctly better performing in a specific parameter setting.

Regarding the kind of pattern that was detected in Section 5.3, that the importance values of the most influential covariates are distinctly higher for the permutation importance only in scenarios combining a sample size of 1000 and correlated covariates, it can not be said that this pattern is in any way reflected by the results of the comparison of correlations or AUCs. Neither does this pattern seem to be reflected in the median differences between correlations/AUCs, which are depicted in Figure 15, where the combination of $n = 1000$ and $rho = 0.5$ does not indicate particularly high or low differences in either direction.

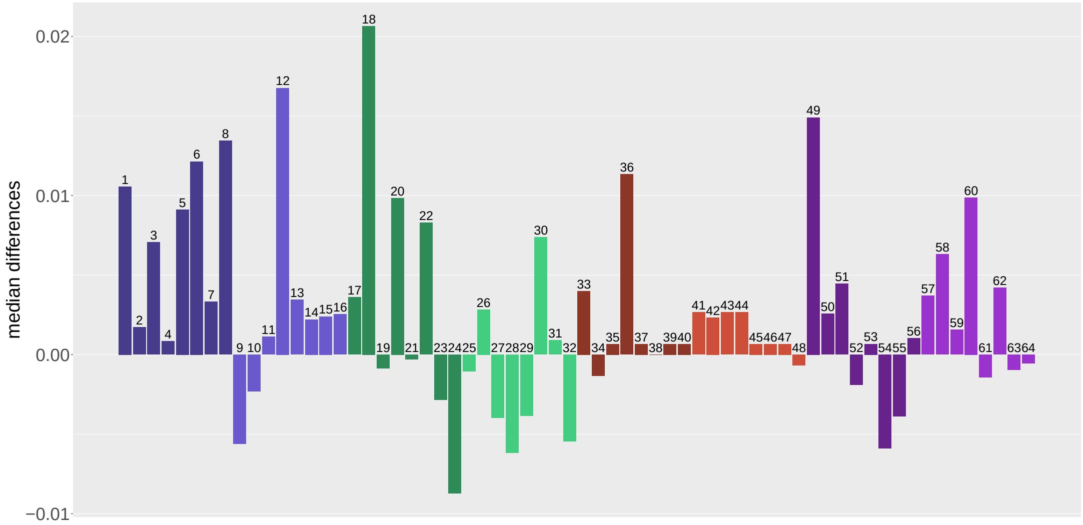


Figure 15: Median differences in correlations/AUCs for all scenarios. Differences where taken in the form of PropRandom - permutation, so that a positive value indicates higher correlations/AUCs for the PropRandom importance. Numbers above bars indicate scenario.

The median differences between correlations or AUCs were calculated by taking only those pairs into account where the difference is unequal to zero, as already explained in the end of Section 5.2. This modification plays only a role in the scenarios that have constant effect sizes as well as correlation between covariates as well as a sample size of 1000, in all other scenarios the differences of all 100 pairs of correlations/AUCs are unequal to zero. The differences were taken by subtracting the permutation importance correlation/AUC from the PropRandom correlation/AUC, which means a positive median difference indicates that the PropRandom correlations/AUCs are higher. The colour scheme in the figure is supposed to give some orientation as to what bars belong to which parameter settings. Green bars belong to scenarios with effects $\boldsymbol{\beta}_I$, blue bars to scenarios with effects $\boldsymbol{\beta}_{II}$, red bars to scenarios with effects $\boldsymbol{\beta}_{III}$, and purple bars to scenarios with effects $\boldsymbol{\beta}_{IV}$. For each colour, the darker shade represents scenarios with no correlation between covariates and the lighter shade represents scenarios with $\rho = 0.5$.

What can be seen in Figure 15 quite clearly is that there are distinctly more median differences which are positive (44 of 64) and that the largest absolute values of the positive differences are much larger than those of the negative differences. This tendency, that in most cases the performance of the PropRandom importance measure is slightly better than that of the permutation importance measure, could already be seen in the analyses of the correlations and AUCs in Sections 5.1 and 5.2 and is here condensed into one figure. Yet there is no pattern recognizable that any parameter setting or combination of settings decisively influences the difference in the performances of the both variable importance measures.

6 Conclusion and outlook

Hapfelmeier et al. (2014) stipulate several requirements that their new variable importance measure should meet. The first of these requirements is: “When there are no missing values, the measure should provide the same variable importance ranking as the original permutation importance measure.” (p. 25) This condition is satisfied in the simulation study they conduct, and the results of this thesis confirm this assessment. How small the actual differences between the importance values of the two measures are in the simulation study done in this thesis is illustrated by Figures 2 and 3 in the beginning of Section 5.1 as well as in Figure 14 in Section 5.3. Given the similarity in the procedure of computing variable importance values for both methods, very similar results for the use of either measure could be expected.

It is therefore not surprising that the performance of both measures is also extremely similar, as can be seen in Figures 4 - 11 in Sections 5.1 and 5.2. Even though a few of the differences between correlations/AUCs could be shown to be significant, it is questionable whether this differences can be considered as relevant, since they are so small. However, the performance of the new PropRandom variable importance is in most cases evaluated as (if only very slightly) higher than that of the original permutation importance, and in the fewer cases where the performance of the permutation importance is the stronger one, the differences are on average even smaller (cf. Figure 15 in Section 5.4). So what can be concluded from the results of the simulation study is that the PropRandom procedure evidently is at least not inferior to the permutation importance in the considered simulation settings. If there is no great advantage in choosing the PropRandom importance over the permutation importance in the absence of missing values, there is also apparently no disadvantage in doing so. Consequently, since the new method additionally has the upside of being able to deal easily with missing data, which the permutation importance lacks, there is no reason why the PropRandom importance should not replace the permutation importance as the default variable importance measure, as it already does in the `R` package `party`.

Of course, it can not be ruled out that there may be other data situations where either of both methods may more distinctly outperform the other. The performance of the two measures could for example also be examined in data situations with a more complex correlation structure, or covariates that are non-normally distributed or categorical, and lots of other settings can be thought of. Moreover the PropRandom procedure might be combined with other adaptations of the original permutation importance. Hapfelmeier et al. (2014) already state that they plan to extend their approach to the conditional permutation importance developed by Strobl et al. (2008), where the covariate values are not permuted within the whole OOB-sample but within a grid of the covariate space that is defined by other covariates that are correlated to the covariate of interest. It is also easy to imagine how the PropRandom procedure may be combined with the AUC-based variable importance that was introduced by Janitza et al. (2013) and was shown to be better suited to strongly imbalanced data situations than the original permutation importance. Such extensions of the PropRandom measure to other importance measures would then also have to be evaluated in different data situations.

Literature

Breiman, L., J. H. Friedman, R. A. Olshen & C.J. Stone (1984). Classification and Regression Trees. Boca Raton u.a.: Chapman & Hall/CRC.

Breiman, L. (1996). Bagging Predictors. *Machine Learning* 24:2. 123–140.

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

Cutler, A., D. R. Cutler & J. R. Stevens (2012). Random Forests. *Ensemble Machine Learning: Methods and Applications.* New York: Springer. 157-176.

Dobra, A. & J. Gehrke (2001). Bias correction in classification tree construction. *Proceedings of the Eighteenth International Conference on Machine Learning.* 90–97. San Mateo: Morgan Kaufmann Publishers Inc. http://www.cs.cornell.edu/people/dobra/papers/icml2001-bias.pdf

Fawcett, T. (2006). An Introduction to ROC Analysis. *Pattern Recognition Letters* 27. 861–874.

Frederic, P. & F. Lad (2003). A Technical Note on the Logitnormal Distribution. https://www.math.canterbury.ac.nz/research/ucdms2003n7.pdf

Genuer, R., J.-M. Poggi & C. Tuleau-Malot. (2008). Random forests: some methodological insights. *Rapport de recherche* RR-6729. INRIA.

Hanley, J. A. & B. J. McNeil (1982). The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve. *Radiology* 143(1). 29-36.

Hapfelmeier, A., T. Hothorn & K. Ulm (2012). Random forest variable importance with missing data. *Technical Report Number 121, Department of Statistics, University of Munich.* https://epub.ub.uni-muenchen.de/12757/1/TechnicalReport_LMU.pdf

Hapfelmeier, A., T. Hothorn, K. Ulm & C. Strobl (2014). A new variable importance measure for random forests with missing data. *Statistical Computing* 24. 21–34.

Hastie, T., R. Tibshirani & J. H. Friedman (2009). *The elements of statistical learning: Data mining, inference and prediction* (Second edition, corrected 7th printing ed.). Springer series in statistics. New York: Springer.

Hollander, M.& D. A. Wolfe (1973). *Nonparametric Statistical Methods.* New York: John Wiley & Sons.

Hornung, R. & A.-L. Boulesteix (2021). Interaction Forests: Identifying and exploiting interpretable quantitative and qualitative interaction effects. *Technical Report No. 237, Department of Statistics, University of Munich.* (Supplementary Material 1)
https://www.ibe.med.uni-muenchen.de/organisation/mitarbeiter/070_drittmittel/hornung/interactionforest_suppfiles/suppmat1_hornungboulesteix2021.pdf

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

Hothorn, T., K. Hornik, C. Strobl & A. Zeileis (2021). *party: A Laboratory for Recursive Partytioning.* R package version 1.3-9.

Janitza, S., C. Strobl & A.-L. Boulesteix (2013). An AUC-based variable importance measure for random forests. *BMC Bioinformatics* 14:119.

Janitza, S., E. Celik & A.-L. Boulesteix (2018). A computationally fast variable importance test for random forests for high-dimensional data. *Advances in Data Analysis and Classification* 12. 885–915.

Loh, W.-Y. & Y.-S. Shih YS (1997). Split Selection Methods for Classification Trees. *Statistica Sinica* 7. 815–840.

Probst, P., A.-L. Boulesteix & B. Bischl (2019). Tunability: Importance of hyperparameters of machine learning algorithms. *The Journal of Machine Learning Research.* 20(53). 1934–1965.

Strobl, C., A.-L. Boulesteix & T. Augustin (2007). Unbiased split selection for classification trees based on the Gini index. *Computational Statistics & Data Analysis* 52(1). 483–501.

Strobl, C., A.-L. Boulesteix, T. Kneib, T. Augustin & A. Zeileis (2008). Conditional variable importance for random forests. *BMC Bioinformatics* 9(1). 307-317.

Strobl, C., J. Malley & G. Tutz (2009). An Introduction to Recursive Partitioning: Rationale, Application and Characteristics of Classification and Regression Trees, Bagging and Random Forests. *Psychological Methods.* 14(4). 323–348.

White, A. & W. Liu (1994). Bias in information based measures in decision tree induction. *Machine Learning* 15(3). 321–329.

White, I. R., P. Royston & A. M. Wood (2011). Multiple imputation using chained equations: Issues and guidance for practice. *Statistics in Medicine.* 30(4). 377–399.

Wutzler, T. (2021). *logitnorm: Functions for the Logitnormal Distribution.* R package version 0.8.38.

Yan, Y. (2016). *MLmetrics: Machine Learning Evaluation Metrics.* R package version 1.1.1.

Appendix

A Additional figures and tables

A.1 Performance evaluation of all simulation settings

			mtry = 21		mtry = 8	
effects	corr.	n	full trees	3 layers	full trees	3 layers
$\boldsymbol{\beta}_I$	$\rho = 0$	200	0.824	0.805	0.835	0.830
$\boldsymbol{\beta}_I$	$\rho = 0$	1000	0.893	0.847	0.916	0.895
$\boldsymbol{\beta}_I$	$\rho = 0.5$	200	0.895	0.885	0.916	0.912
$\boldsymbol{\beta}_I$	$\rho = 0.5$	1000	0.939	0.895	0.955	0.936
$\boldsymbol{\beta}_{II}$	$\rho = 0$	200	0.855	0.850	0.867	0.870
$\boldsymbol{\beta}_{II}$	$\rho = 0$	1000	0.911	0.880	0.937	0.921
$\boldsymbol{\beta}_{II}$	$\rho = 0.5$	200	0.917	0.916	0.939	0.931
$\boldsymbol{\beta}_{II}$	$\rho = 0.5$	1000	0.951	0.920	0.966	0.952
$\boldsymbol{\beta}_{III}$	$\rho = 0$	200	0.772	0.759	0.784	0.792
$\boldsymbol{\beta}_{III}$	$\rho = 0$	1000	0.863	0.808	0.887	0.852
$\boldsymbol{\beta}_{III}$	$\rho = 0.5$	200	0.783	0.770	0.794	0.795
$\boldsymbol{\beta}_{III}$	$\rho = 0.5$	1000	0.838	0.804	0.849	0.829
$\boldsymbol{\beta}_{IV}$	$\rho = 0$	200	0.776	0.769	0.790	0.788
$\boldsymbol{\beta}_{IV}$	$\rho = 0$	1000	0.871	0.837	0.894	0.882
$\boldsymbol{\beta}_{IV}$	$\rho = 0.5$	200	0.870	0.862	0.880	0.882
$\boldsymbol{\beta}_{IV}$	$\rho = 0.5$	1000	0.927	0.893	0.940	0.928

Table 4: Performance for all 64 simulation settings, measured with the AUC.

A.2 Proportions of the response classes in data simulated with effects $\boldsymbol{\beta}^*_{IV}$

corr.	n	proportion of $y = 1$ in data
$\rho = 0$	200	0.49932
$\rho = 0$	1000	0.49970
$\rho = 0.5$	200	0.50004
$\rho = 0.5$	1000	0.49996

Table 5: Proportions of the response classes in data simulated with effects $\boldsymbol{\beta}^*_{IV}$. Proportions are averaged over 10000 simulated data sets.

A.3 Figures of the variable importance values of all scenarios

Effects: β_I, no correlation between covariates, $n = 200$

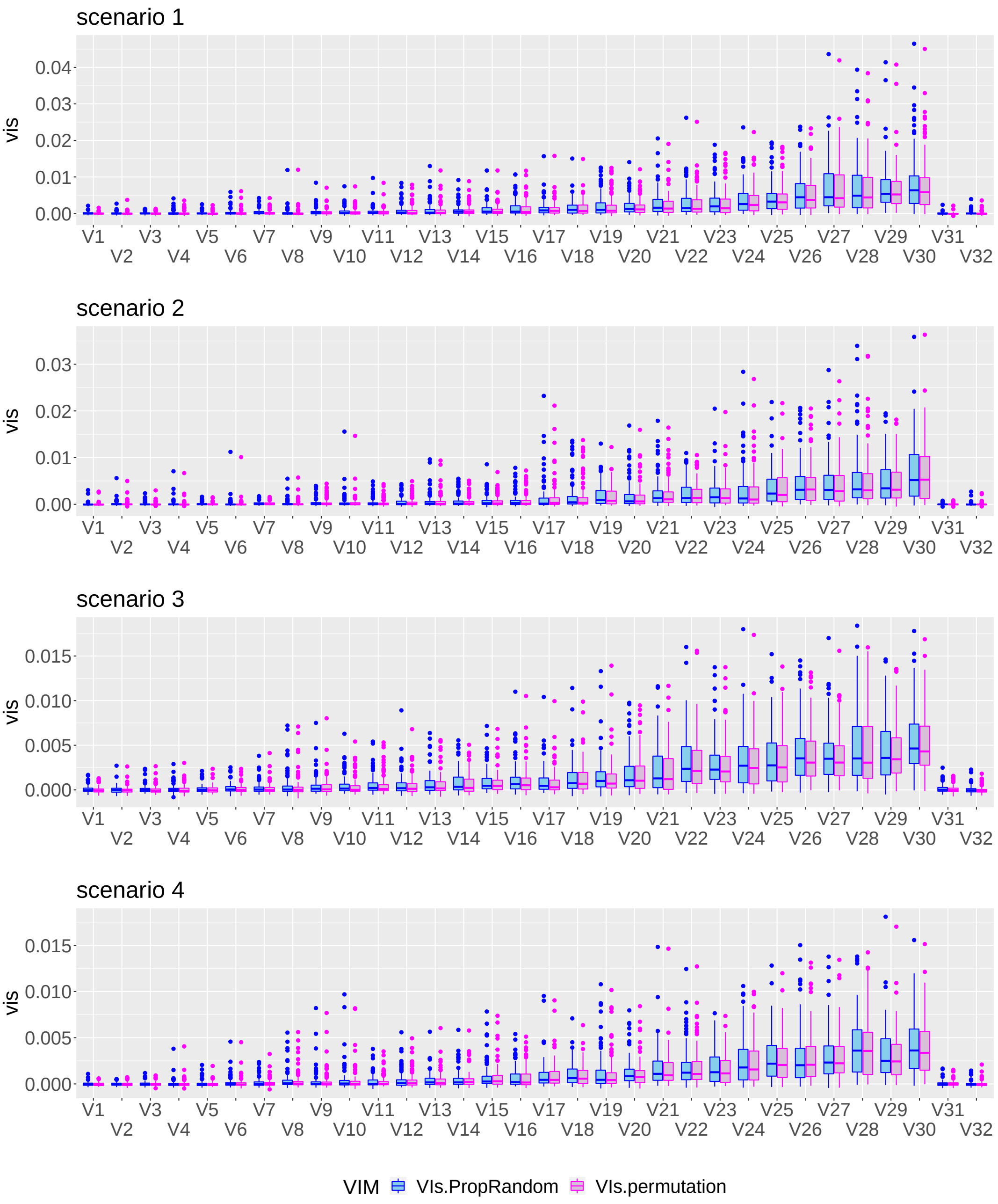


Figure 16: PropRandom (blue) and permutation (pink) variable importance values for scenarios 1 - 4. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

Effects: β_I, no correlation between covariates, $n = 1000$

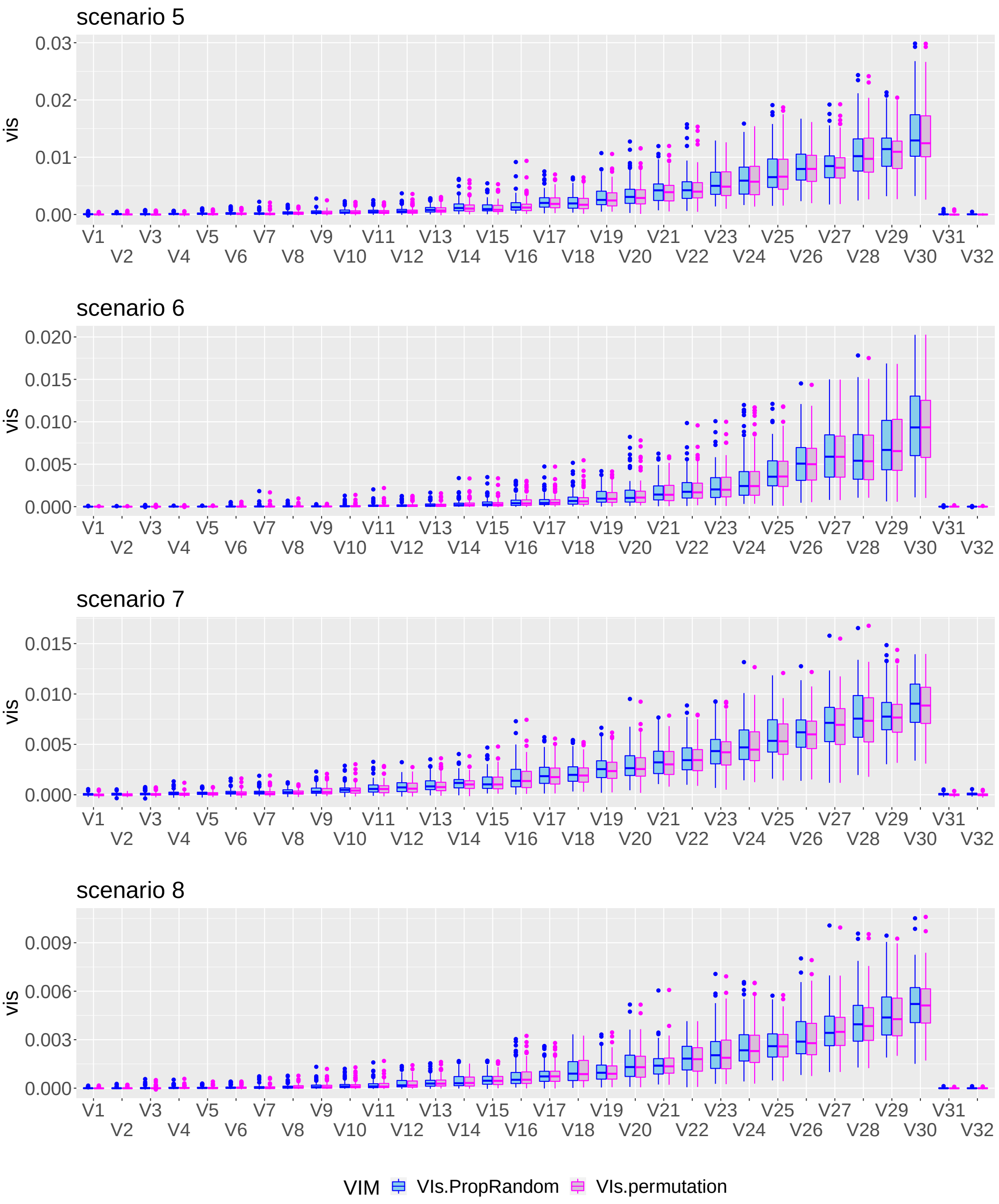


Figure 17: PropRandom (blue) and permutation (pink) variable importance values for scenarios 5 - 8. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

Effects: β_I, blockwise correlation between covariates, $n = 200$

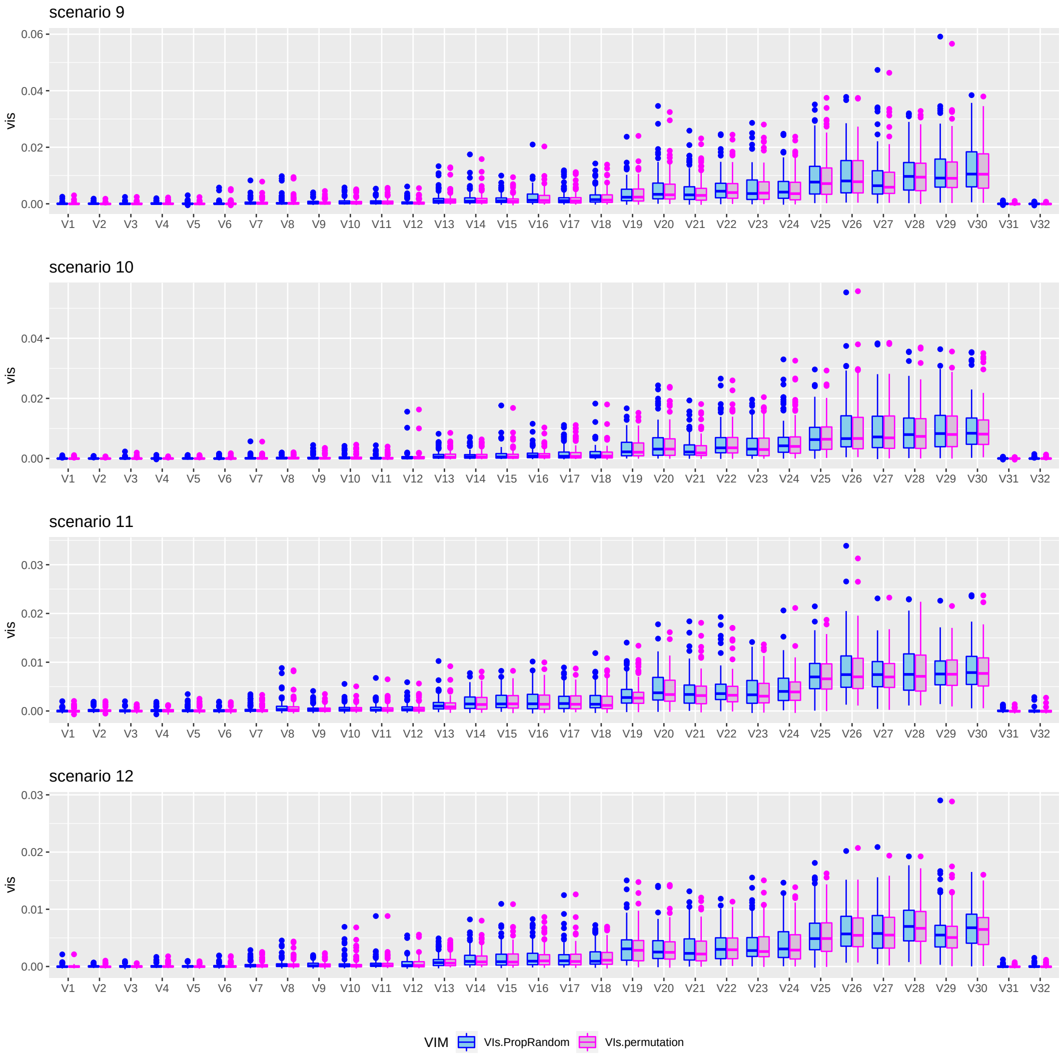


Figure 18: PropRandom (blue) and permutation (pink) variable importance values for scenarios 9 - 12. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

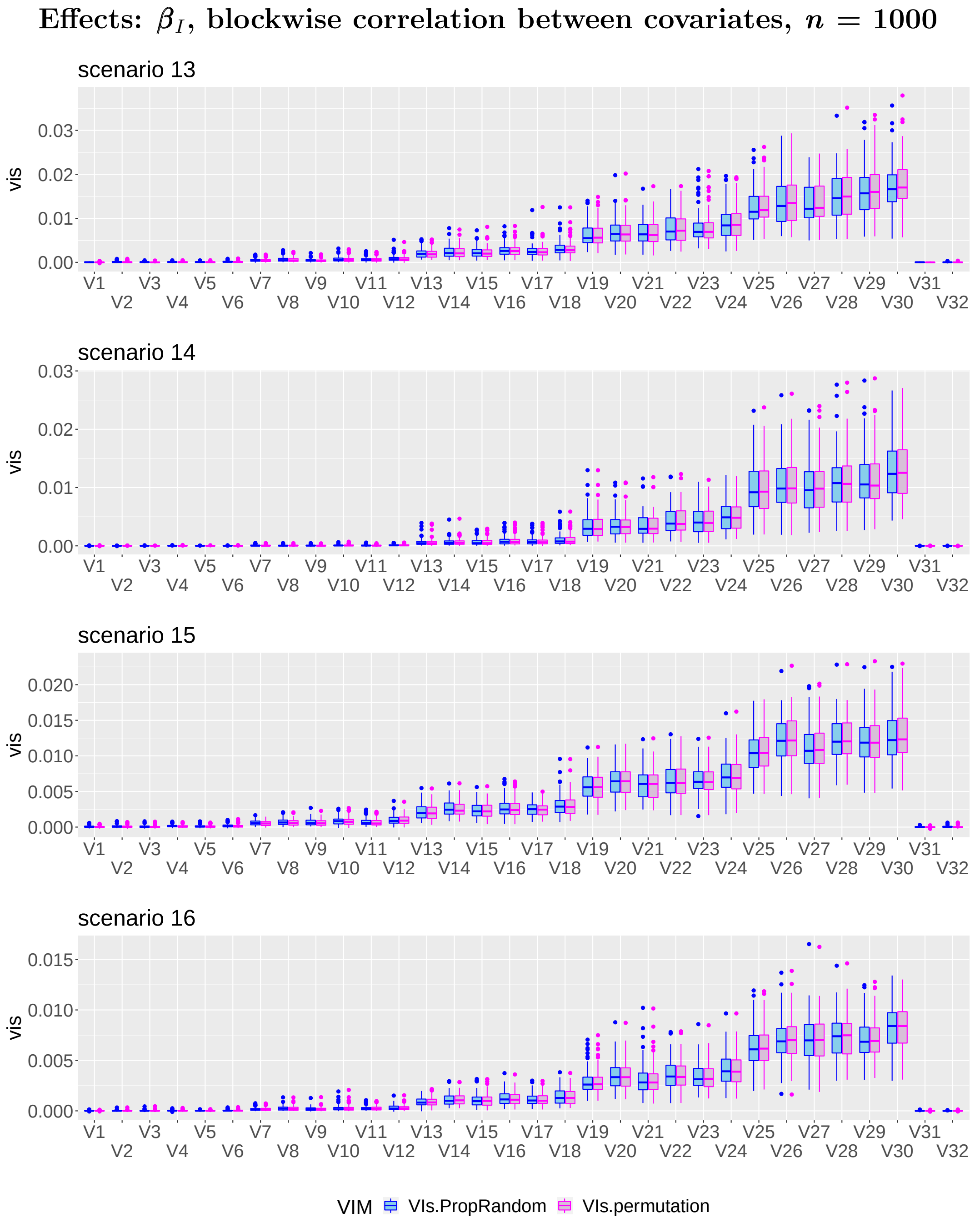


Figure 19: PropRandom (blue) and permutation (pink) variable importance values for scenarios 13 - 16. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

Effects: β_{II}, no correlation between covariates, n = 200

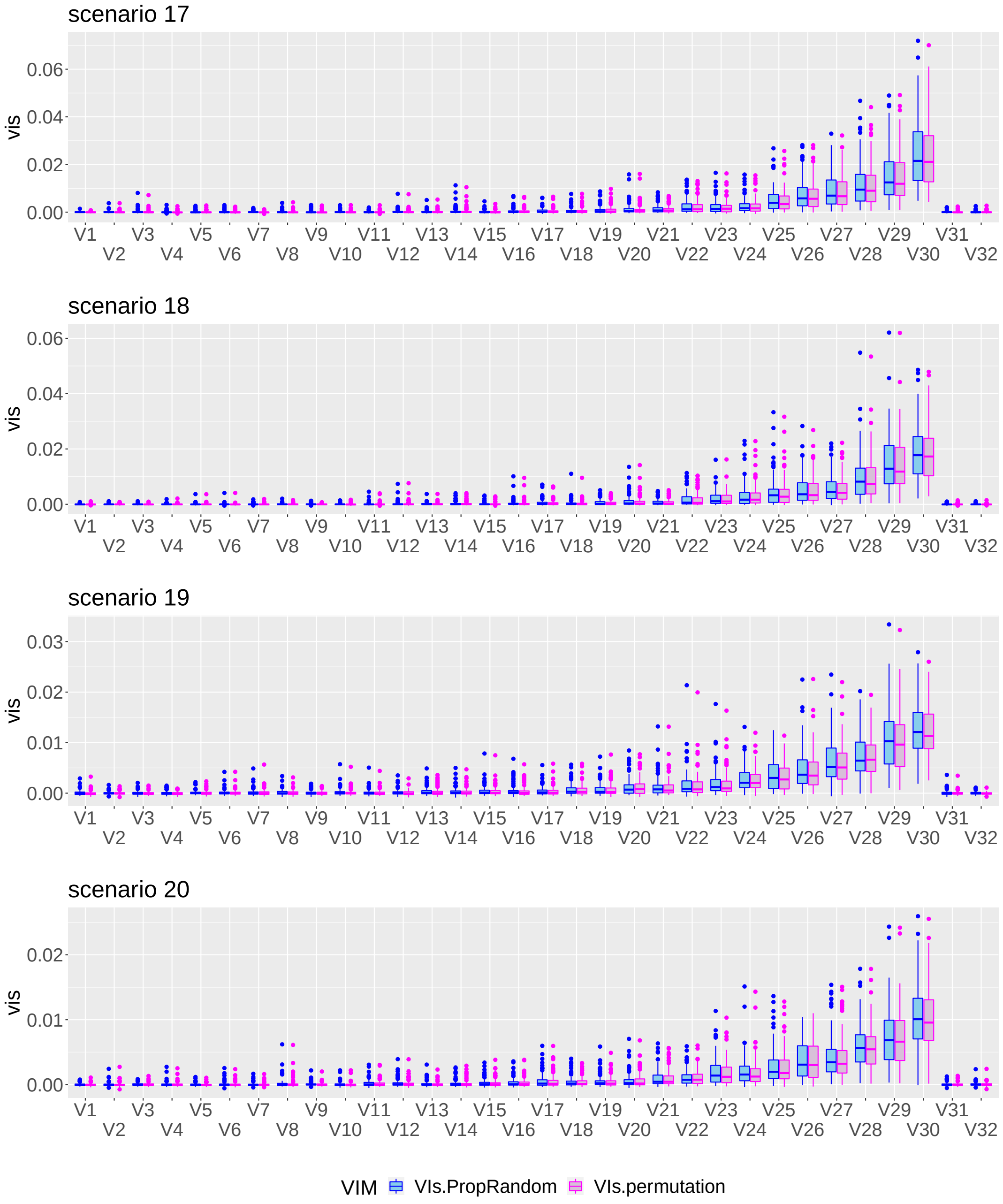


Figure 20: PropRandom (blue) and permutation (pink) variable importance values for scenarios 17 - 20. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

Effects: β_{II}, no correlation between covariates, n = 1000

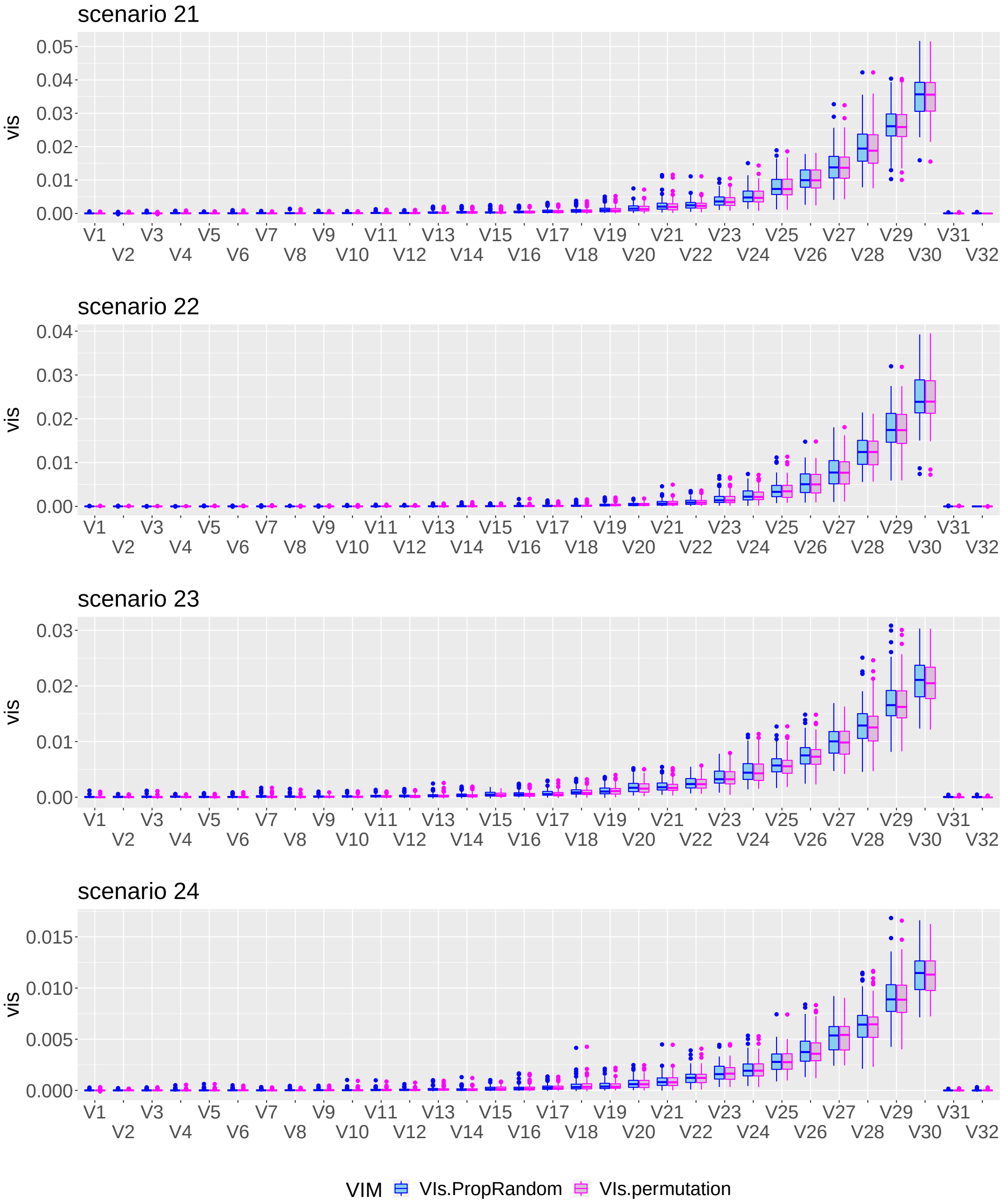


Figure 21: PropRandom (blue) and permutation (pink) variable importance values for scenarios 21 - 24. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

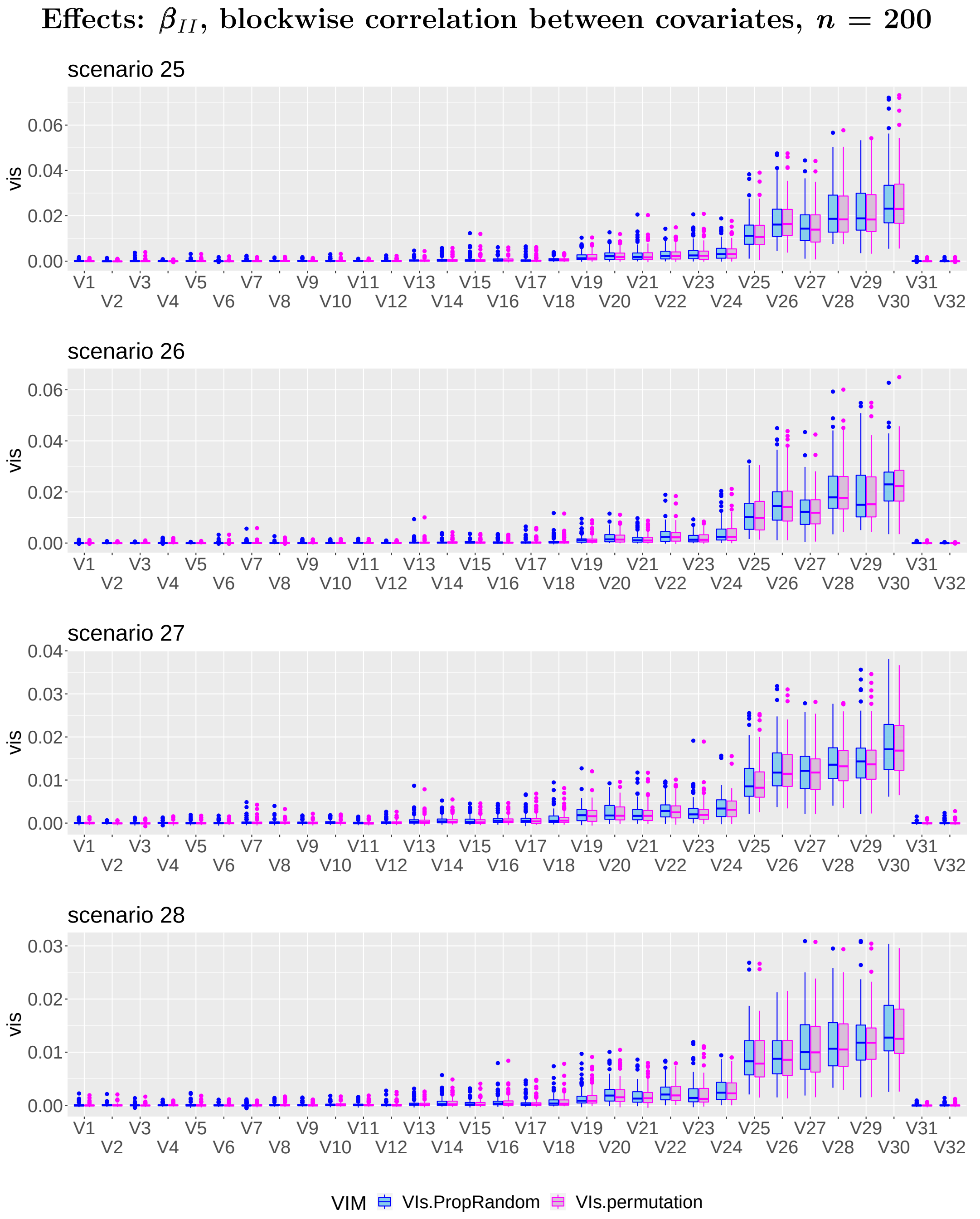


Figure 22: PropRandom (blue) and permutation (pink) variable importance values for scenarios 25 - 28. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

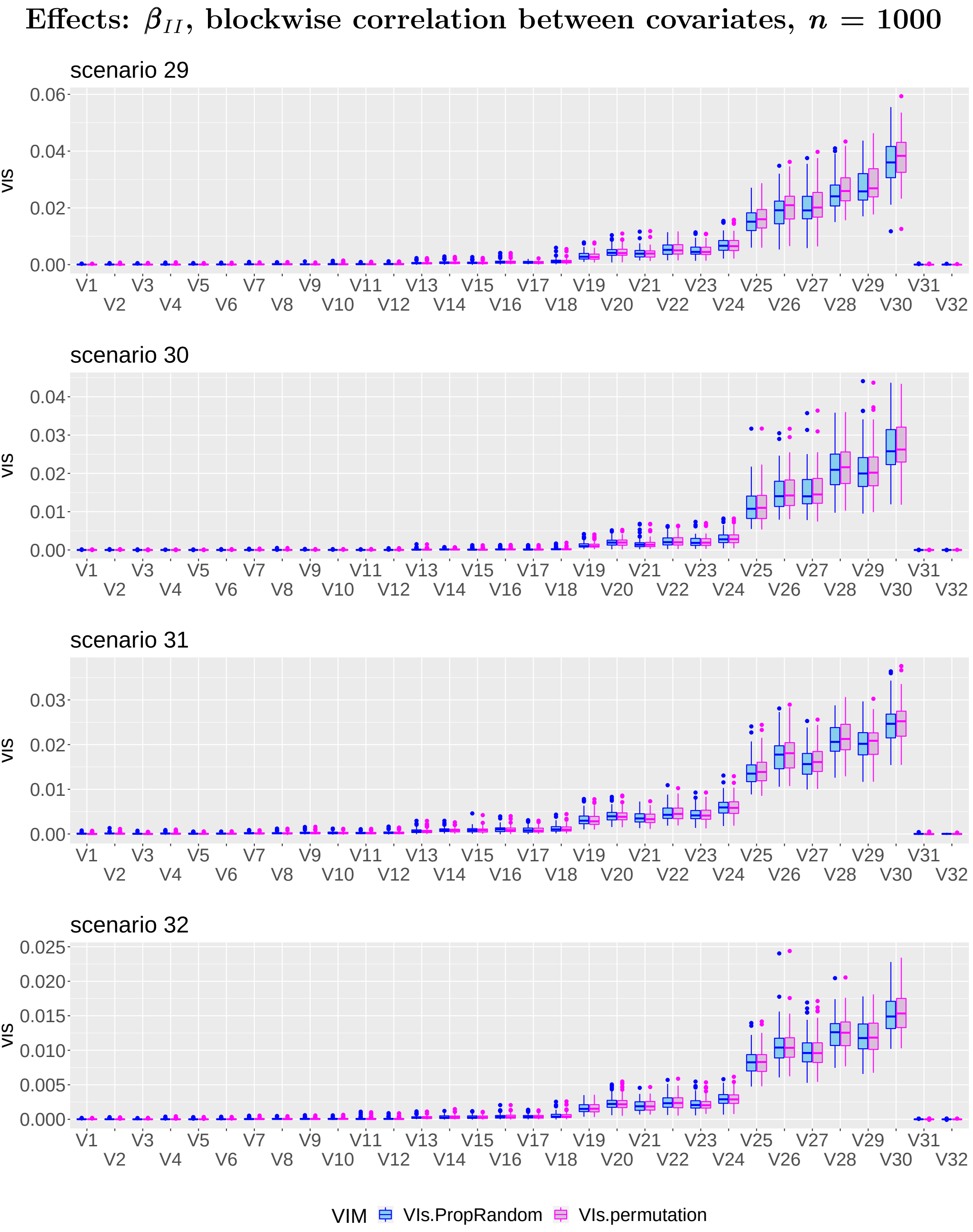


Figure 23: PropRandom (blue) and permutation (pink) variable importance values for scenarios 29 - 32. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

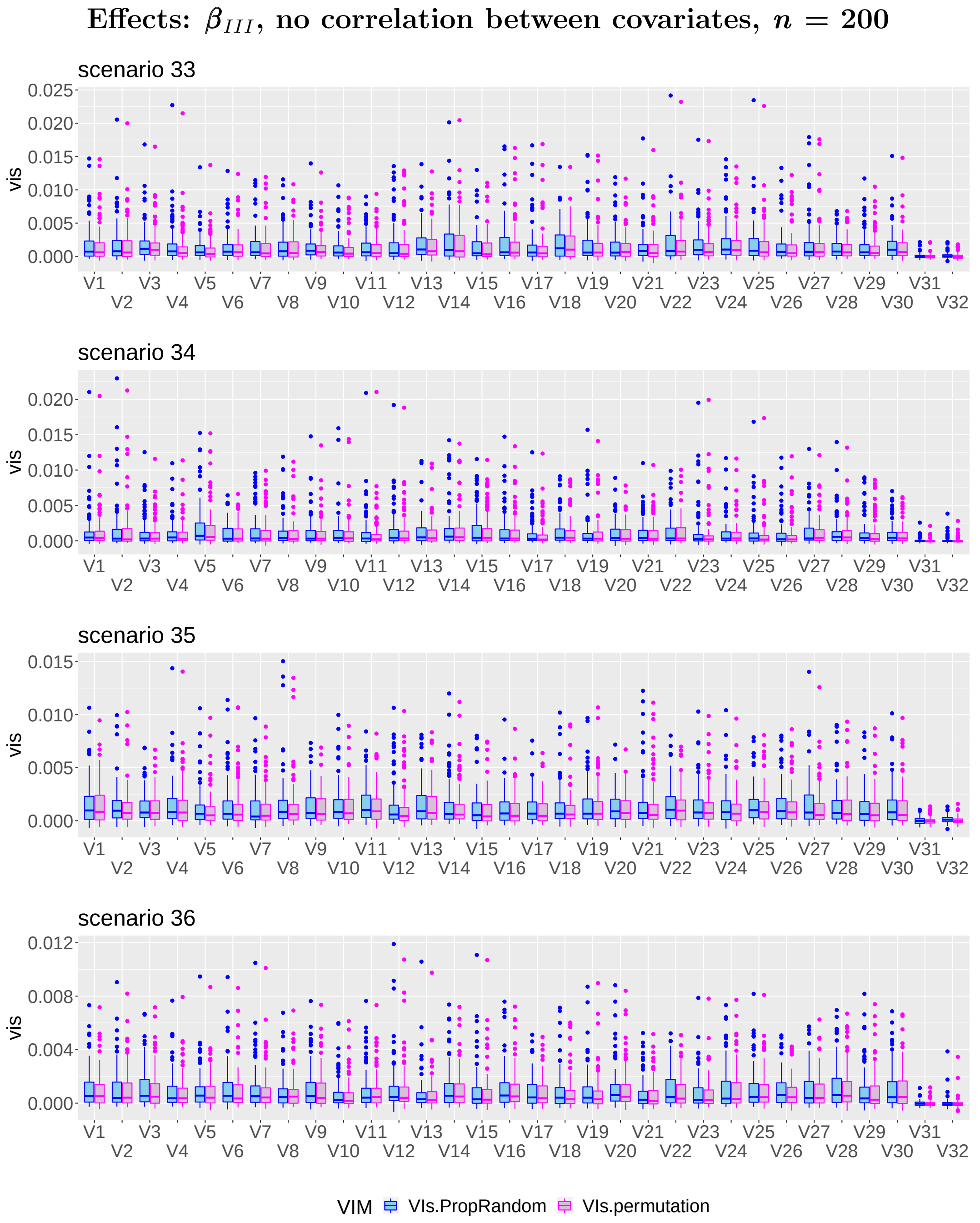


Figure 24: PropRandom (blue) and permutation (pink) variable importance values for scenarios 33 - 36. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

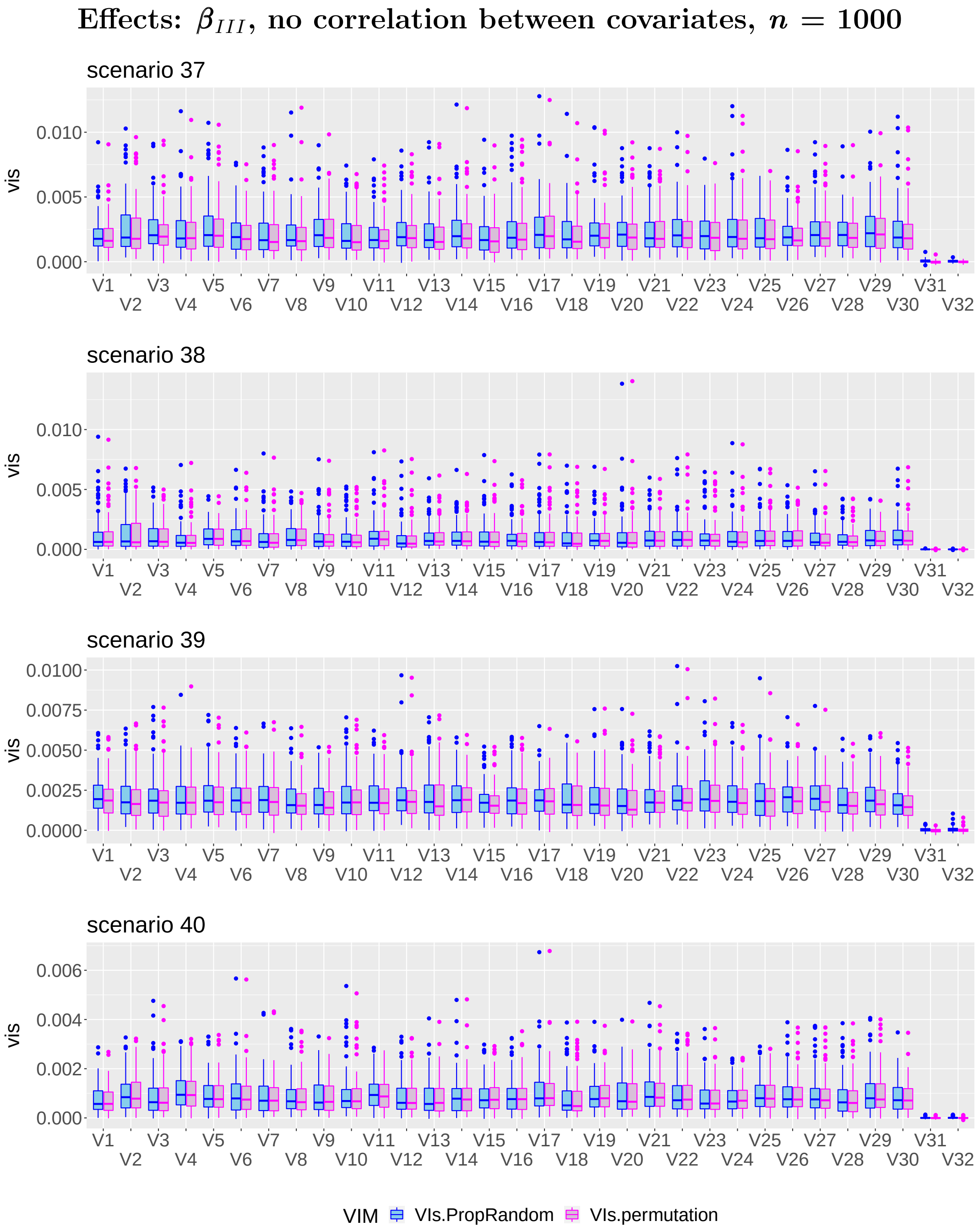


Figure 25: PropRandom (blue) and permutation (pink) variable importance values for scenarios 37 - 40. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

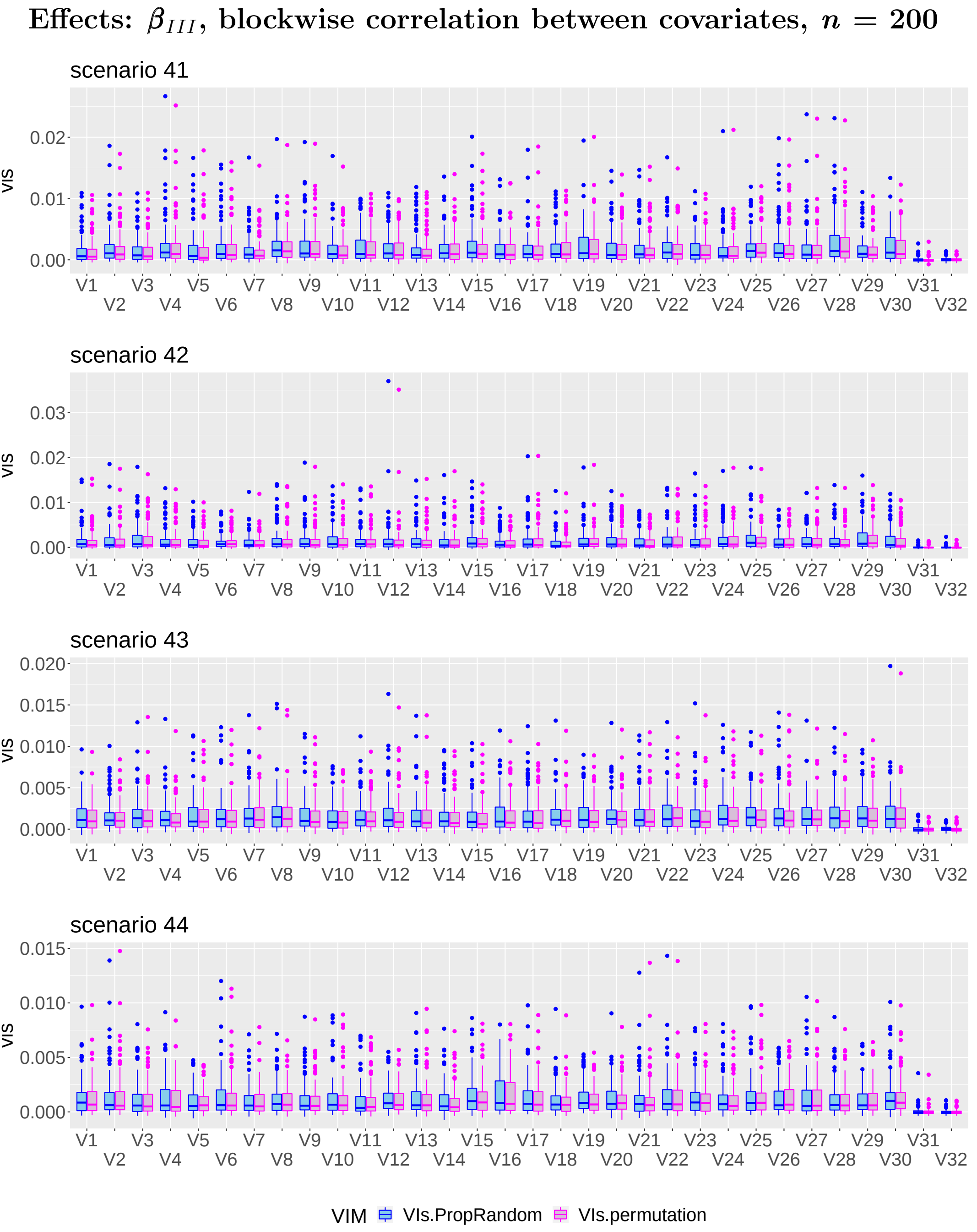


Figure 26: PropRandom (blue) and permutation (pink) variable importance values for scenarios 41 - 44. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

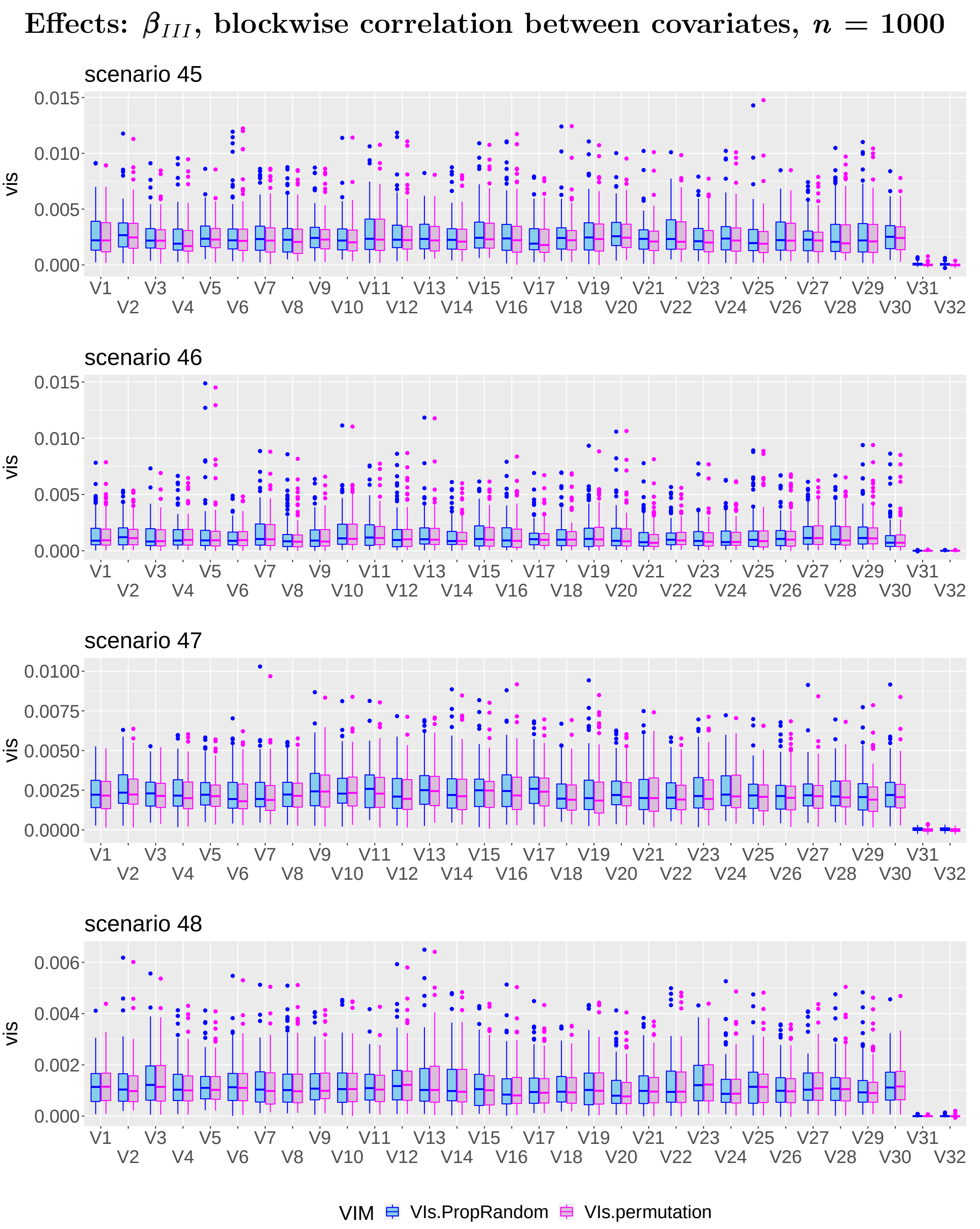


Figure 27: PropRandom (blue) and permutation (pink) variable importance values for scenarios 45 - 48. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

Effects: β_{IV}, no correlation between covariates, $n = 200$

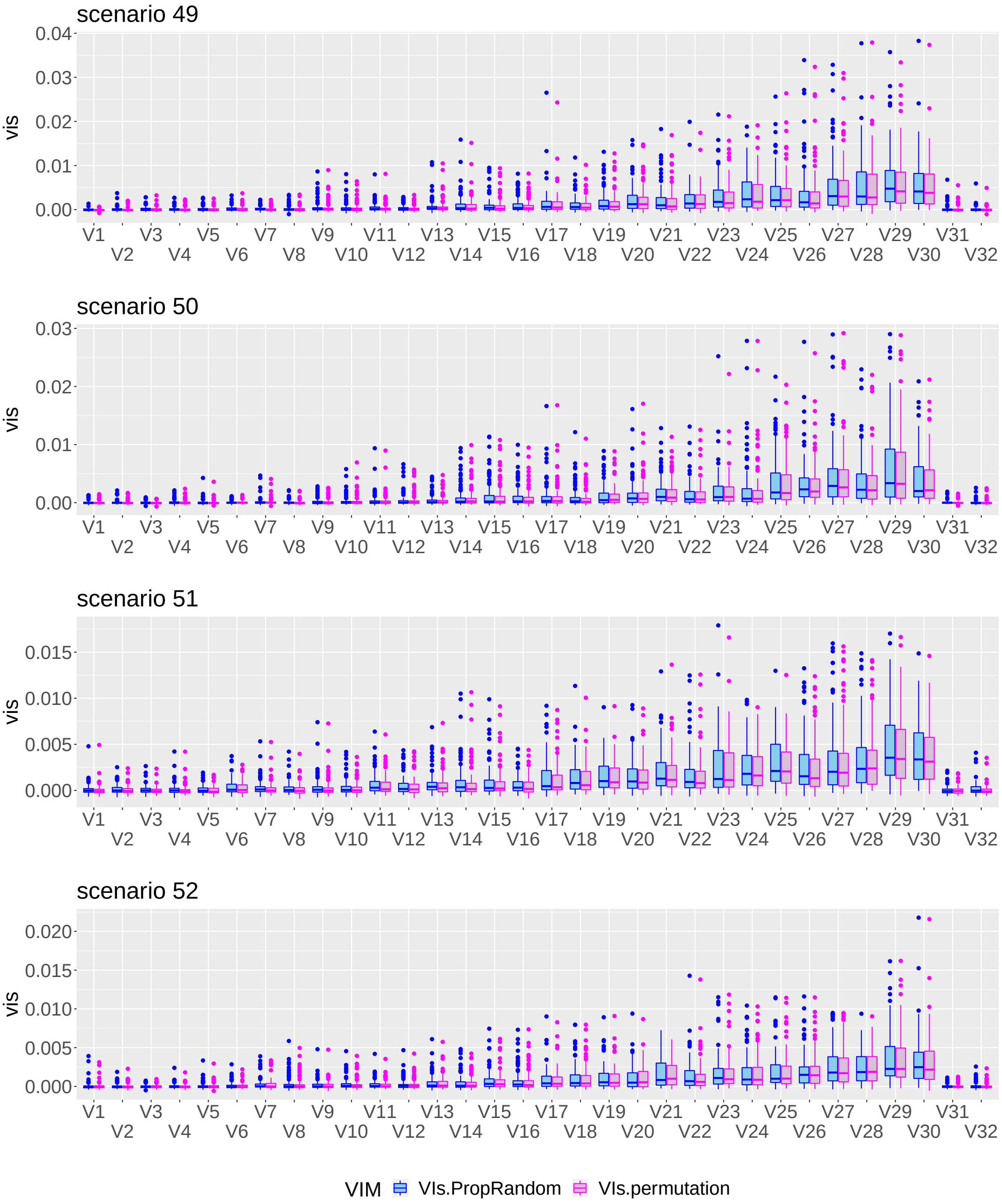


Figure 28: PropRandom (blue) and permutation (pink) variable importance values for scenarios 49 - 52. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

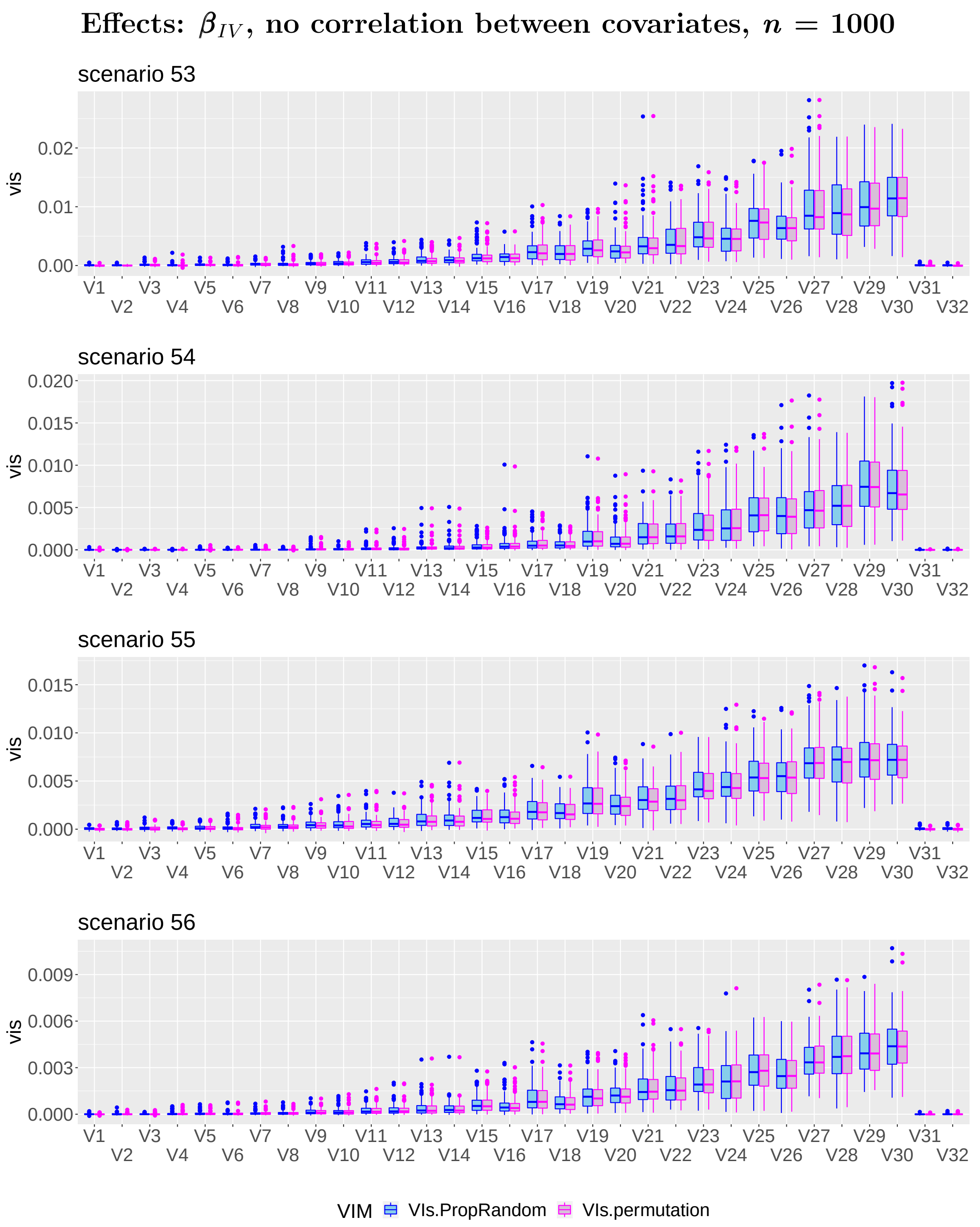


Figure 29: PropRandom (blue) and permutation (pink) variable importance values for scenarios 53 - 56. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

Effects: β_{IV}, blockwise correlation between covariates, $n = 200$

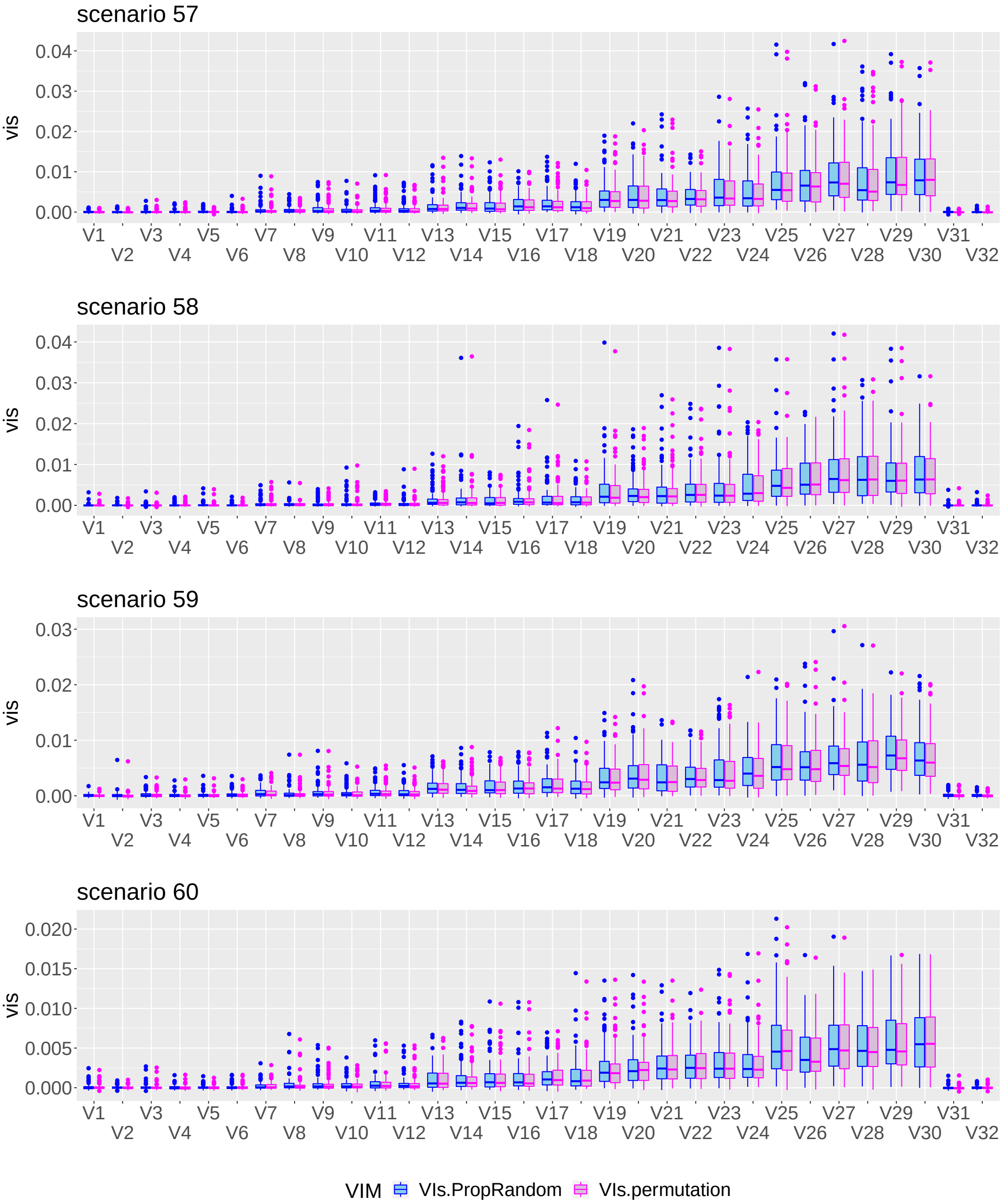


Figure 30: PropRandom (blue) and permutation (pink) variable importance values for scenarios 57 - 60. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

Effects: β_{IV}, blockwise correlation between covariates, $n = 1000$

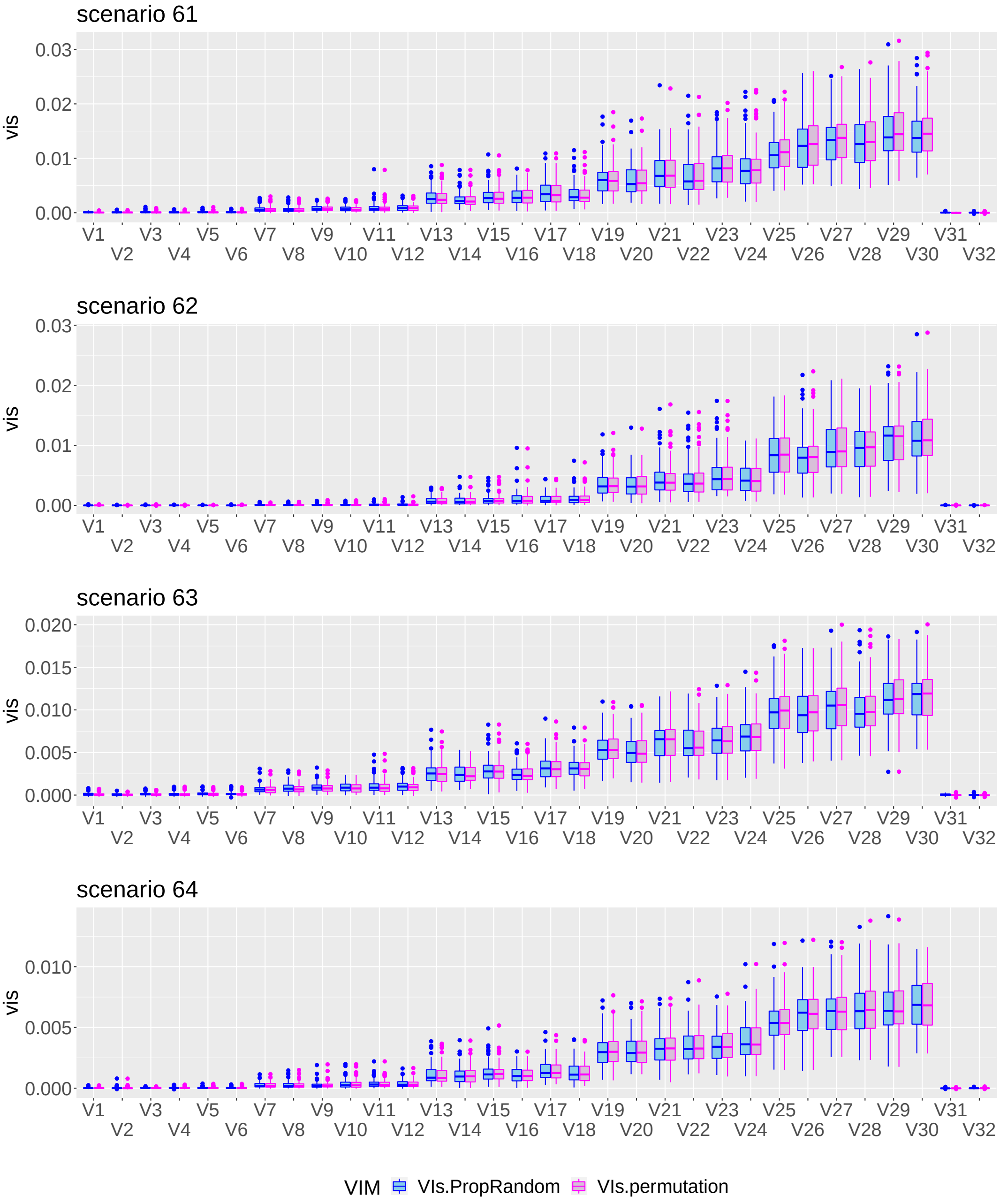


Figure 31: PropRandom (blue) and permutation (pink) variable importance values for scenarios 61 - 64. Each boxplot comprises 100 values for the corresponding variable. Variables 33 - 80 are excluded for clarity.

A.4 Differences between the avaraged VIs of the influential covariates

Effects: β_I, no correlation between covariates

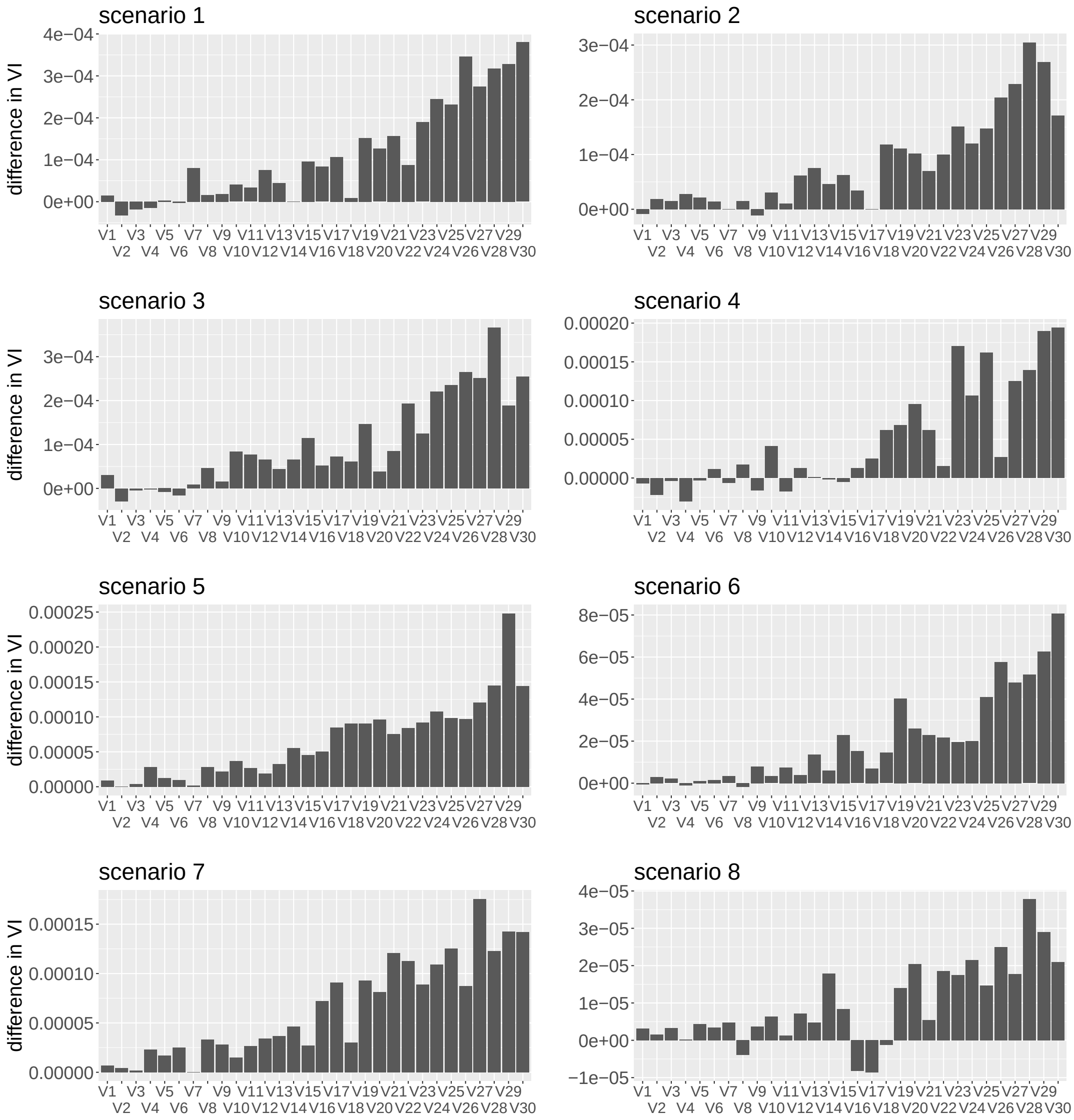


Figure 32: Differences between average PropRandom and permutation VIs for scenarios 1 - 8. Scenarios in the upper half have a sample size of 200, scenarios in the lower half have a sample size of 1000. $maxdepth = 0$ for scenarios on the left side and 3 for scenarios on the right side. Scenarios in the same row share the same value for $mtry$, alternating 21 and 8.

Effects: $\boldsymbol{\beta}_I$, blockwise correlation between covariates

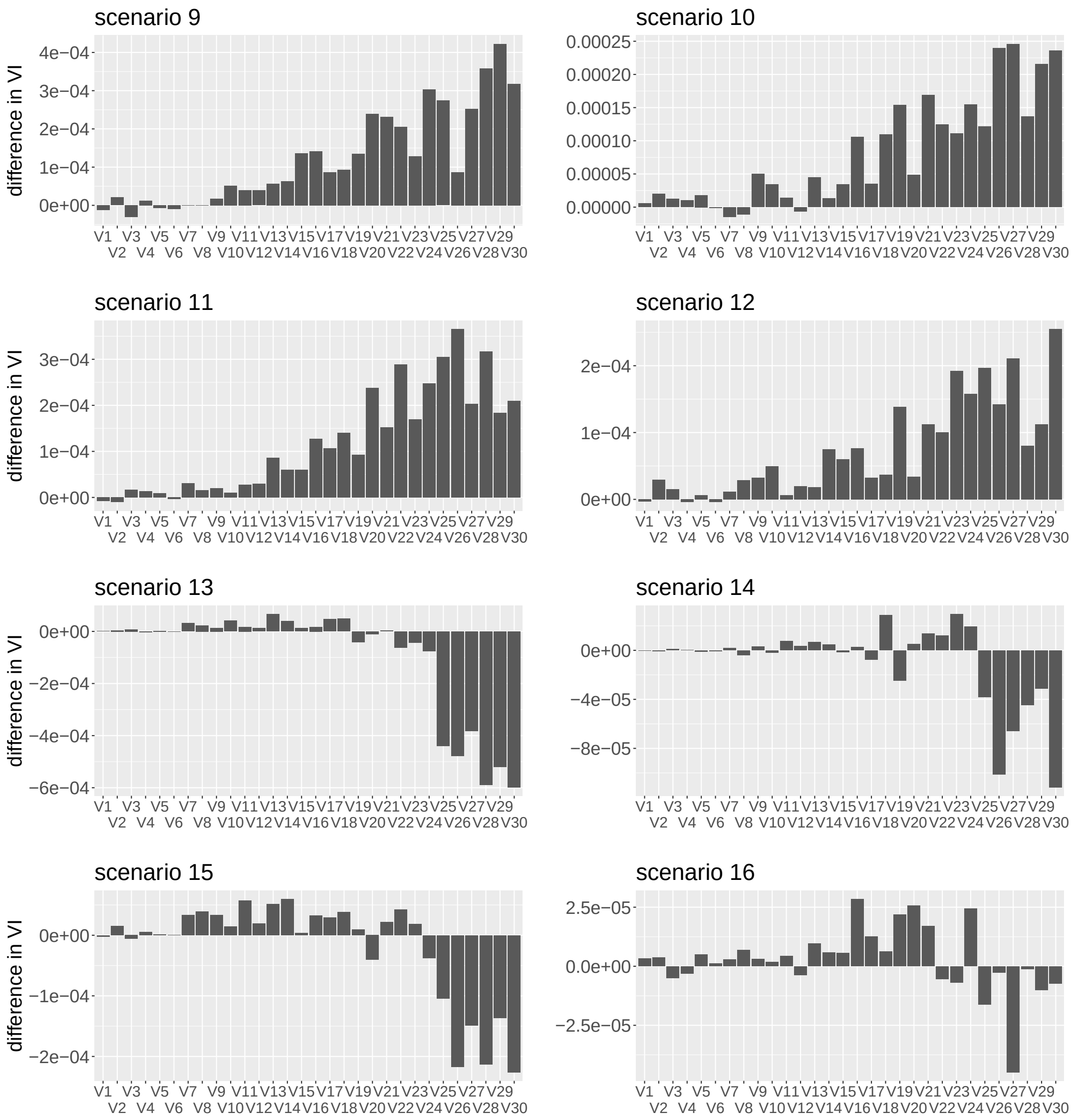


Figure 33: Differences between average PropRandom and permutation VIs for scenarios 9 - 16. Scenarios in the upper half have a sample size of 200, scenarios in the lower half have a sample size of 1000. $maxdepth = 0$ for scenarios on the left side and 3 for scenarios on the right side. Scenarios in the same row share the same value for $mtry$, alternating 21 and 8.

Effects: β_{II}, no correlation between covariates

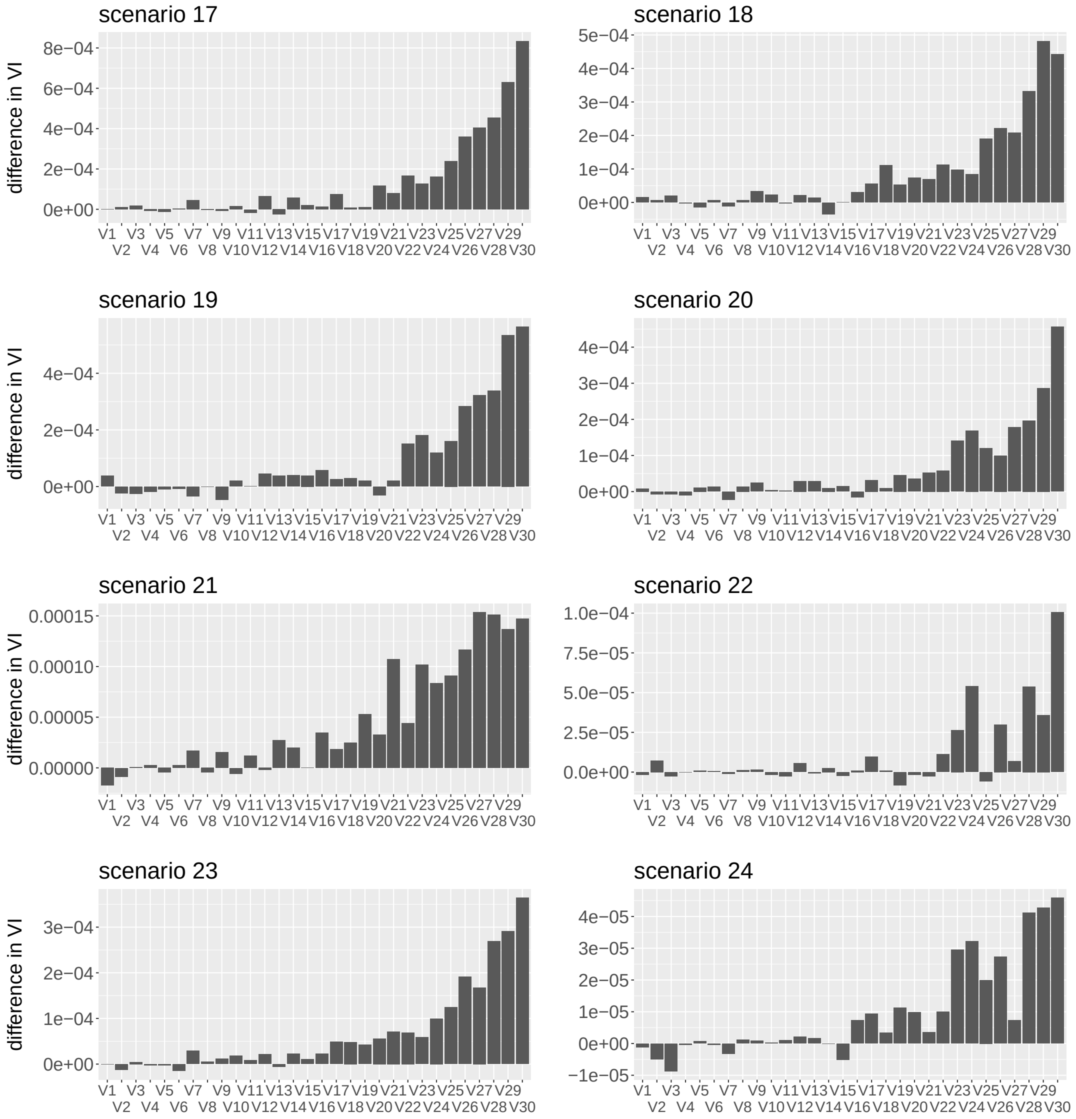


Figure 34: Differences between average PropRandom and permutation VIs for scenarios 17 - 24. Scenarios in the upper half have a sample size of 200, scenarios in the lower half have a sample size of 1000. $maxdepth = 0$ for scenarios on the left side and 3 for scenarios on the right side. Scenarios in the same row share the same value for $mtry$, alternating 21 and 8.

Effects: β_{II}, blockwise correlation between covariates

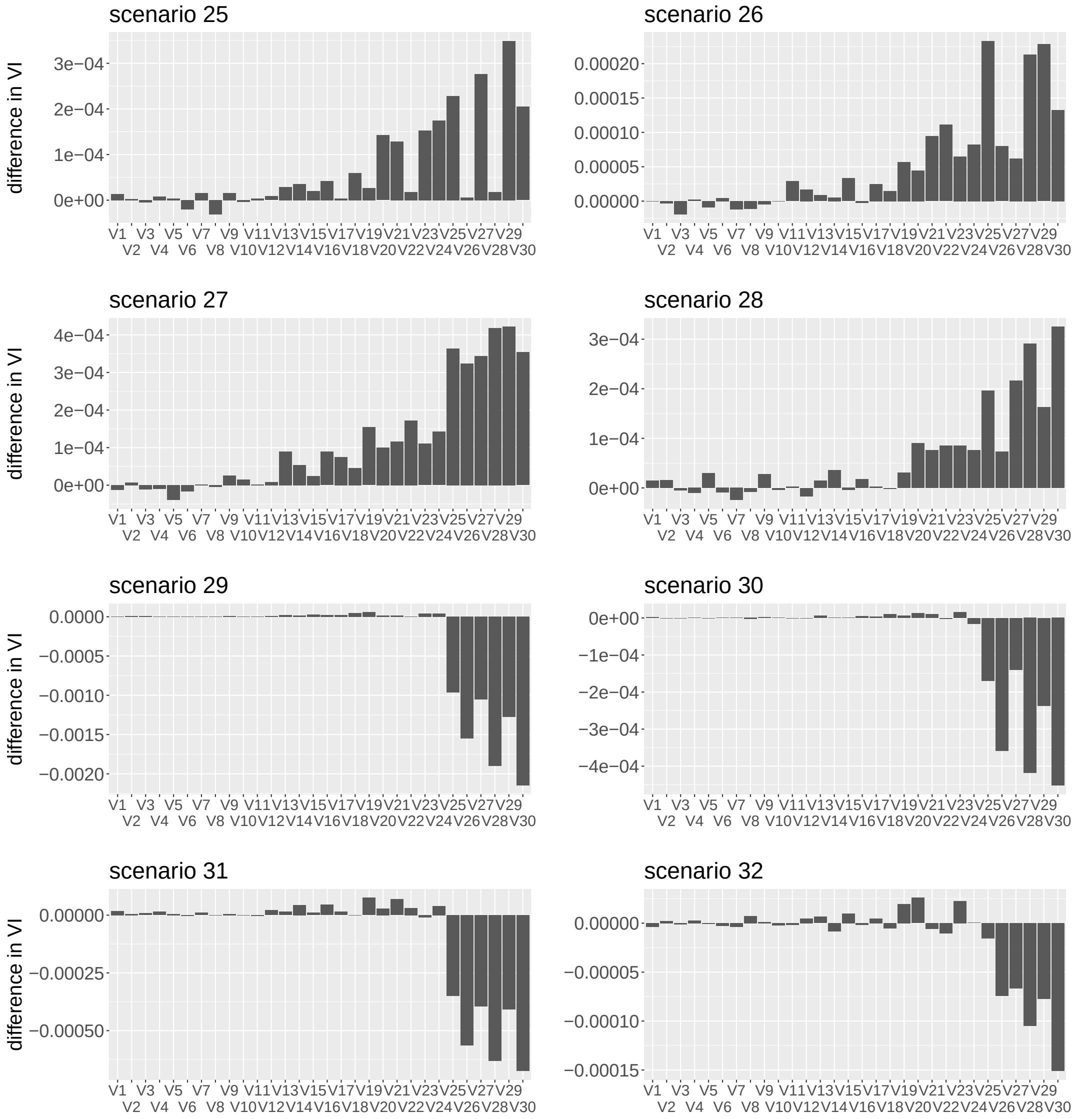


Figure 35: Differences between average PropRandom and permutation VIs for scenarios 25 - 32. Scenarios in the upper half have a sample size of 200, scenarios in the lower half have a sample size of 1000. $maxdepth = 0$ for scenarios on the left side and 3 for scenarios on the right side. Scenarios in the same row share the same value for $mtry$, alternating 21 and 8.

Effects: β_{III}, no correlation between covariates

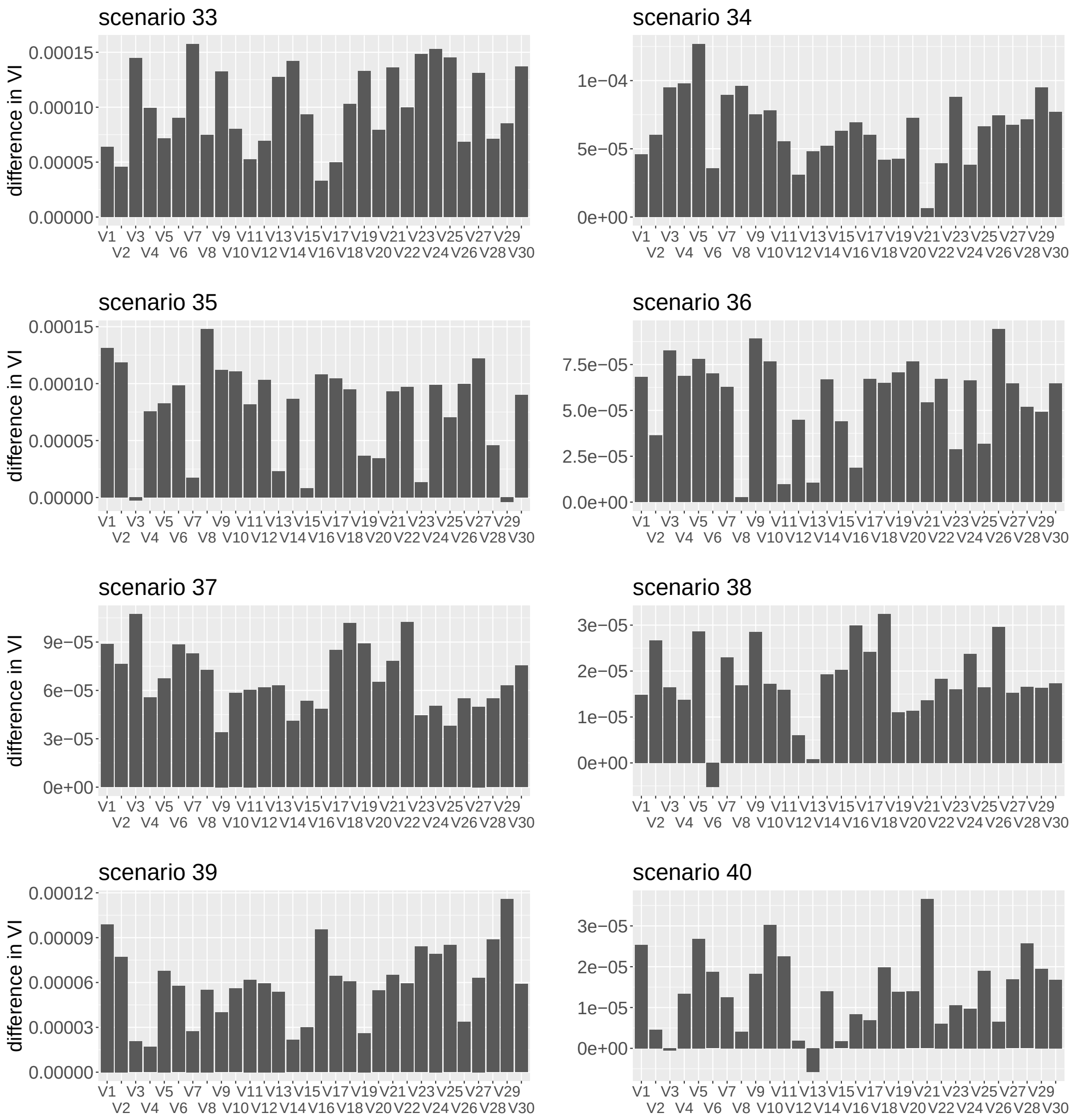


Figure 36: Differences between average PropRandom and permutation VIs for scenarios 33 - 40. Scenarios in the upper half have a sample size of 200, scenarios in the lower half have a sample size of 1000. $maxdepth = 0$ for scenarios on the left side and 3 for scenarios on the right side. Scenarios in the same row share the same value for $mtry$, alternating 21 and 8.

Effects: $\boldsymbol{\beta}_{III}$, blockwise correlation between covariates

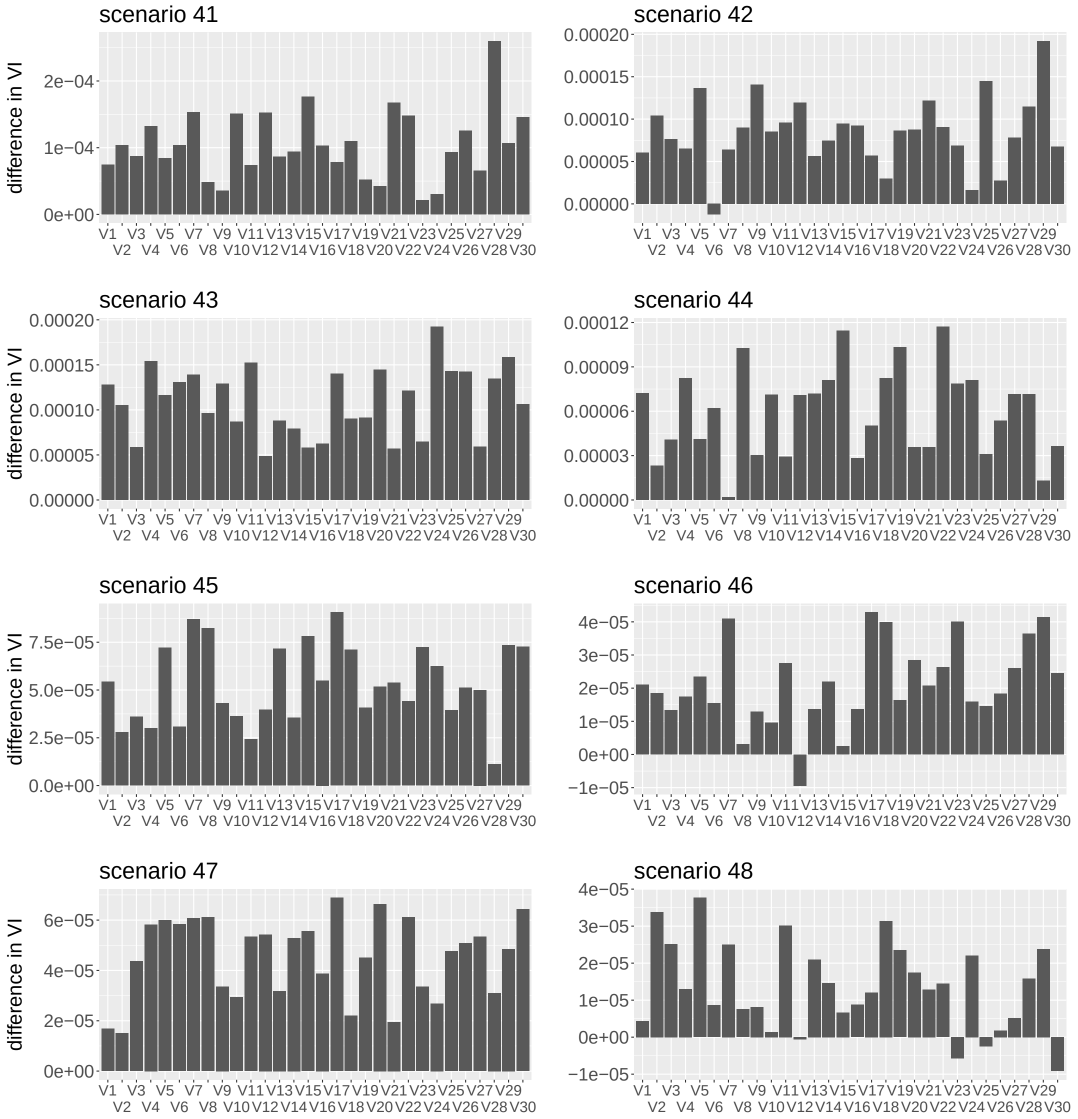


Figure 37: Differences between average PropRandom and permutation VIs for scenarios 41 - 48. Scenarios in the upper half have a sample size of 200, scenarios in the lower half have a sample size of 1000. $maxdepth = 0$ for scenarios on the left side and 3 for scenarios on the right side. Scenarios in the same row share the same value for $mtry$, alternating 21 and 8.

Effects: β_{IV}, no correlation between covariates

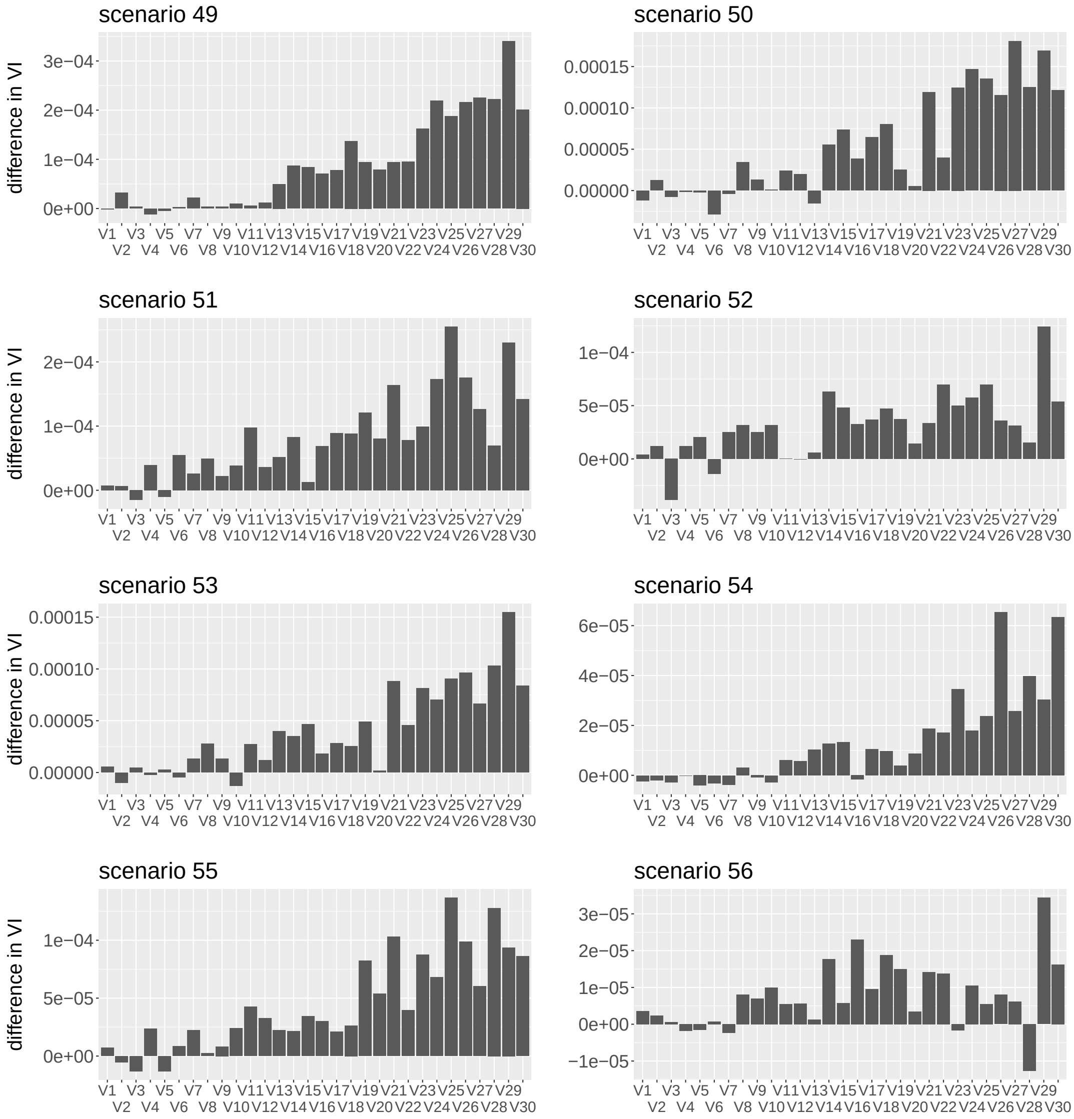


Figure 38: Differences between average PropRandom and permutation VIs for scenarios 49 - 56. Scenarios in the upper half have a sample size of 200, scenarios in the lower half have a sample size of 1000. $maxdepth = 0$ for scenarios on the left side and 3 for scenarios on the right side. Scenarios in the same row share the same value for $mtry$, alternating 21 and 8.

Effects: β_{IV}, blockwise correlation between covariates

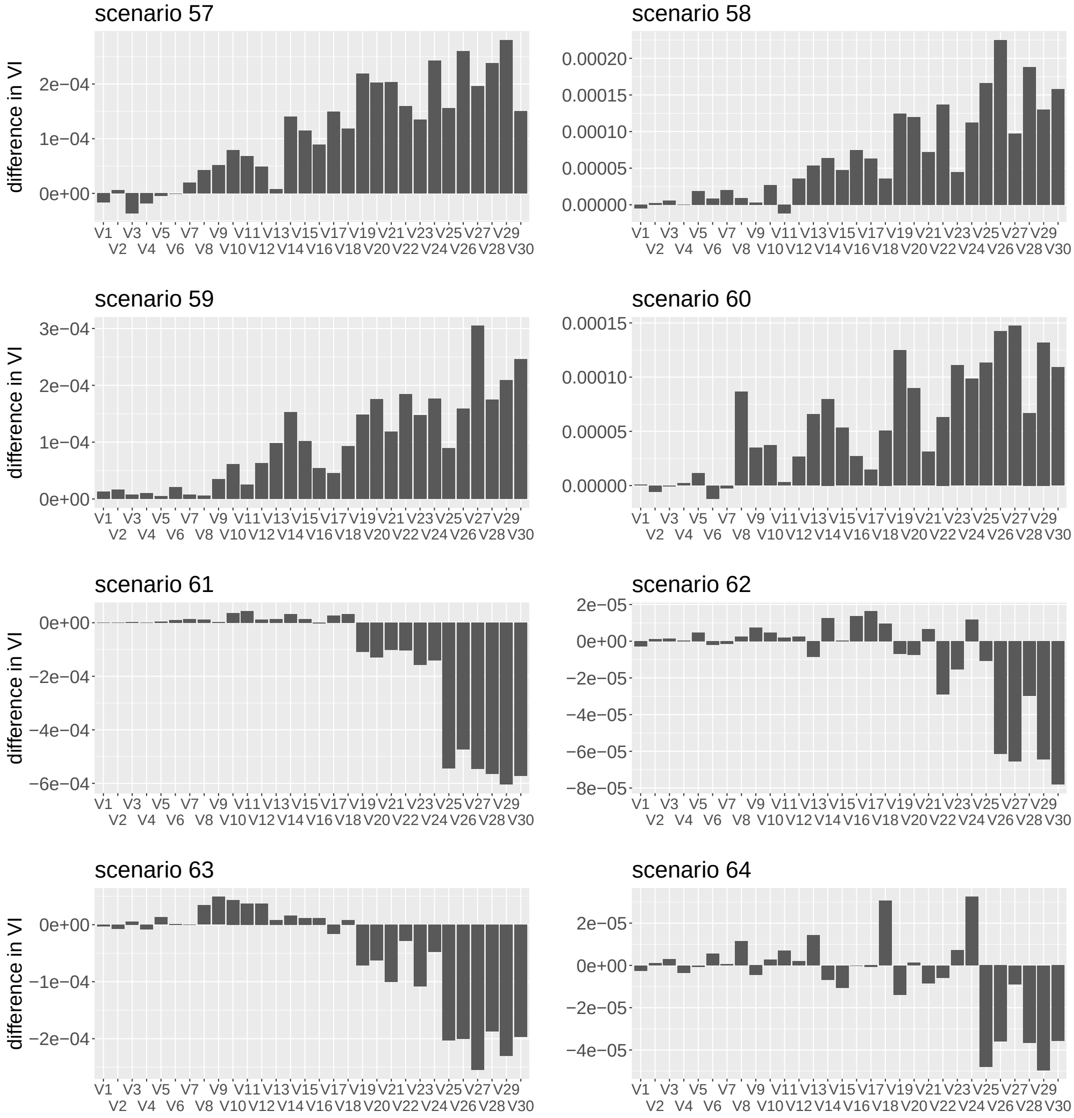


Figure 39: Differences between average PropRandom and permutation VIs for scenarios 57 - 64. Scenarios in the upper half have a sample size of 200, scenarios in the lower half have a sample size of 1000. $maxdepth = 0$ for scenarios on the left side and 3 for scenarios on the right side. Scenarios in the same row share the same value for $mtry$, alternating 21 and 8.

B Electronic appendix

The electronic appendix contains:

- A pdf version of this thesis
- The folder *results* contains the VIs and correlations of each scenario in .RData files and the file *load_results.R* to load the results.
  Each .RData file corresponds to one scenario and is a list containing
  - $vis: an 100x80 array with the variable importance values of this scenario computed with PropRandom
  - $ca: a vector with 100 rank correlation coefficients between the effect sizes and the PropRandom importance values (or 100 AUCs evaluating the PropRandom importance values)
  - $vis_old: an 100x80 array with the variable importance values of this scenario computed with the permutation importance
  - $ca_old: a vector with 100 rank correlation coefficients between the effect sizes and the permutation importance values (or 100 AUCs evaluating the permutation importance values)
- The folder *R_code* contains
  - *params_and_functions.R*, which contains the parameters for the simulation and functions used in the files of this folder
  - *simulation.R*, which contains the code to simulate all scenarios
  - .R files for the values in the tables in the thesis (proportion of response classes, AUCs for evaluating the performance of the RFs, p-values of the Wilcoxon tests)
- The folder figures_pdf contains all figures as pdfs that are used in the thesis (sorted into subfolders that are named according to the sections in which the figures are used)
- The folder *figures_code* contains R files to produce all figures that are named of according to the section in which the figures are used

The following packages have to be installed to run the code:

- party
- mvtnorm

- MLmetrics
- tidyverse
- ggplot2
- patchwork

I declare that I have developed and written the enclosed Master's Thesis completely by myself, and have not used sources or means without declaration in the text. Any thoughts from others or literal quotations are clearly marked. The Master's Thesis was not used in the same or in a similar version to achieve an academic grading or is being published elsewhere.

München, 15.12.2021

Katrin Racic-Rachinsky