Computational Statistics and Data Analysis 72 (2014) 92–104

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

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

Recursive partitioning for missing data imputation in the presence of interaction effects☆

L.L. Doove [a,b,*], S. Van Buuren [c,a], E. Dusseldorp [c,b]

[a] Department of Methodology and Statistics, Faculty of Social Sciences, University of Utrecht, PO Box 80140, 3508 TC Utrecht, The Netherlands
[b] Department of Psychology, Katholieke Universiteit Leuven, Tiensestraat 102 - bus 3713, Leuven, Belgium
[c] Netherlands Organisation for Applied Scientific Research TNO, PO Box 2215, 2301 CE Leiden, The Netherlands

ARTICLE INFO

Article history:
Received 18 July 2012
Received in revised form 31 October 2013
Accepted 31 October 2013
Available online 13 November 2013

Keywords:
CART
Classification and regression trees
Interaction problem
MICE
Nonlinear relations
Random forests

ABSTRACT

Standard approaches to implement multiple imputation do not automatically incorporate nonlinear relations like interaction effects. This leads to biased parameter estimates when interactions are present in a dataset. With the aim of providing an imputation method which preserves interactions in the data automatically, the use of recursive partitioning as imputation method is examined. Three recursive partitioning techniques are implemented in the multiple imputation by chained equations framework. It is investigated, using simulated data, whether recursive partitioning creates appropriate variability between imputations and unbiased parameter estimates with appropriate confidence intervals. It is concluded that, when interaction effects are present in a dataset, substantial gains are possible by using recursive partitioning for imputation compared to standard applications. In addition, it is shown that the potential of recursive partitioning imputation approaches depends on the relevance of a possible interaction effect, the correlation structure of the data, and the type of possible interaction effect present in the data.

1. Introduction

Today's state of the art solution for handling missing data is multiple imputation. In approaches to implement multiple imputation, different methods are available to use the information from the data at hand (Van Buuren, 2012). The common element in these methods is that they model the relations between variables. Hereby, it is particularly important to reflect the structure of the data since otherwise, parameter estimates under multiple imputation will be biased. Caution is therefore needed when data contain nonlinear structures like a quadratic relation. Approaches to implement multiple imputation, like Multiple Imputation by Chained Equations (MICE; Van Buuren, 2007), do not automatically incorporate nonlinear relations. We focus on a special case of nonlinear relations, namely interaction effects. For the purpose of this study, both cross-products and quadratic terms are denoted by interactions.

MICE is a popular approach for implementing multiple imputation because of its flexibility. In MICE, multivariate missing data are imputed on a variable by variable basis, called fully conditional specification (Van Buuren, 2007). This means that per variable imputations are created, such that for each incomplete variable a specified imputation model is required. In these imputation models, interactions can be modelled in two ways: first, by specifying models including interaction effects manually and second by imputing subgroups of the data separately. For example, one could create distinct imputation

☆ Supplementary materials related to the implementation of proposed methods are available online (see Appendix C).

* Corresponding author at: Department of Psychology, Katholieke Universiteit Leuven, Tiensestraat 102 - bus 3713, Leuven, Belgium. Tel.: +32 16 3 25977.
E-mail addresses: lisa.doove@ppw.kuleuven.be (L.L. Doove), stef.vanbuuren@tno.nl (S. Van Buuren), elise.dusseldorp@tno.nl (E. Dusseldorp).

0167-9473/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.csda.2013.10.025

models for males and females. Besides the fact that both approaches are somewhat cumbersome, they are often unusable as the structure of the data is usually unknown before imputation. Therefore, models should preferably be fitted to the data in an automatic fashion without unnecessary user involvement.

A technique that can handle interactions with ease is recursive partitioning (Burgette and Reiter, 2010; Hand, 1997). One of the first implementations of recursive partitioning is called Automatic Interaction Detection (Morgan and Sonquist, 1963). The recursive partitioning technique models the interaction structure in the data by sequentially splitting a dataset into increasingly homogeneous subsets (Breiman et al., 1984). Essentially recursive partitioning finds the split that is most predictive of the response variable by searching through all predictor variables (Merkle and Schaffer, 2011). Within the subgroups created from one predictor variable, the algorithm goes on to partition the data based on other variables or other splits of the same predictor. The resulting series of splits can be represented by a tree structure like Fig. 1, to which we will return in Section 2. Since splits are conditional on previous splits, the variables used may indicate interaction effects. By constructing models in this manner, possible interactions are automatically taken into account.

Others have worked on this idea of combining recursive partitioning with imputation methods, e.g., Burgette and Reiter (2010), Iacus and Porro (2007, 2008), Nonyane and Foulkes (2007), Stekhoven and Bühlmann (2012), and Van Buuren (2012, p. 83). The main shortcoming of most of the proposed methods is that recursive partitioning is combined with single imputation instead of multiple imputation. Therefore, they cannot be used for making appropriate statistical inferences. Another shortcoming is that, except for Burgette and Reiter, the performance of these methods is not investigated on data containing interaction effects. In the current study, we would like to overcome these shortcomings by providing a framework for connecting recursive partitioning techniques with multiple imputation. This type of imputation takes into account the uncertainty associated with the missing data (Rubin, 1996), which results in parameter estimates with appropriate confidence intervals.

The purpose of our study is to gain insight into whether the use of recursive partitioning in multiple imputation (i.e., MICE) is a convenient way to preserve interaction effects. We consider two main questions: which recursive partitioning techniques create appropriate variability between repeated imputations? What are the statistical properties (e.g., bias, coverage, confidence interval width) of estimates of the interaction parameters? In gaining insight into these questions, distinctions will be made between different types of interactions. In addition, the two questions will be considered for both continuous and categorical data. Burgette and Reiter (2010) embarked on the implementation of recursive partitioning in MICE and demonstrated the performance of the method on a single model with continuous predictor and response variables. We want to elaborate on the work of Burgette and Reiter and, to be complete, also consider categorical predictor and response variables. Different results are expected for both types of data since recursive partitioning techniques are known to perform especially well for data with interactions between categorical variables (Dusseldorp et al., 2010).

This paper is organized as follows. In Section 2, MICE will first be elaborated further after which two main recursive partitioning techniques will be considered, namely Classification And Regression Trees (CART; Breiman et al., 1984) and random forests (Breiman, 2001). Subsequently, incorporation of recursive partitioning in the MICE framework will be presented. In Section 3 different interaction types will be discussed, which will be observed in answering the research questions. Then we make the distinction between predictor and response variables either being continuous (Section 4) or categorical (Section 5). In both Sections 4 and 5, a simulation study is described, carried out to investigate which of the discussed methods are convenient to preserve interaction effects, followed by the results of the simulation study. The results from both simulation studies will be discussed in Section 6, at the end of which some final conclusions are given.

2. MICE and recursive partitioning

2.1. Multiple imputation by chained equations

Imagine a set of variables, $y_1, \ldots, y_j$, some or all of which have missing values. Handling these data using MICE comprises three main steps: generating multiple imputation, analyzing the imputed data, and pooling the analysis results (Van Buuren, 2007). The main idea is to impute each incomplete variable using its own imputation model. All missing values are initially filled in at random. The first variable with at least one missing value, say y_1, is then regressed on the remaining variables, $y_2, \ldots, y_j$. This is restricted to individuals with observed values for y_1. The missing values in y_1 are now replaced by simulated draws from the posterior predictive distribution of y_1. The next variable with missing values, say y_2, is then regressed on all the other variables, $y_1, y_3, \ldots, y_j$. This estimation is restricted to individuals with observed y_2 and uses the imputed values of y_1. Again, missing values in y_2 are replaced by draws from the posterior predictive distribution of y_2. This process is repeated for all other variables with missing values in turn. To stabilize the results this cycle is iterated a number of times, producing one imputed dataset. The entire procedure is repeated m times, yielding m imputed datasets. Each complete dataset is analyzed separately by MICE, after which the results are pooled.

2.2. Recursive partitioning

In this study we consider two main recursive partitioning techniques, namely CART and random forests. We will first elaborate on CART and return to random forests later on in this section. Depending on the response variable of interest

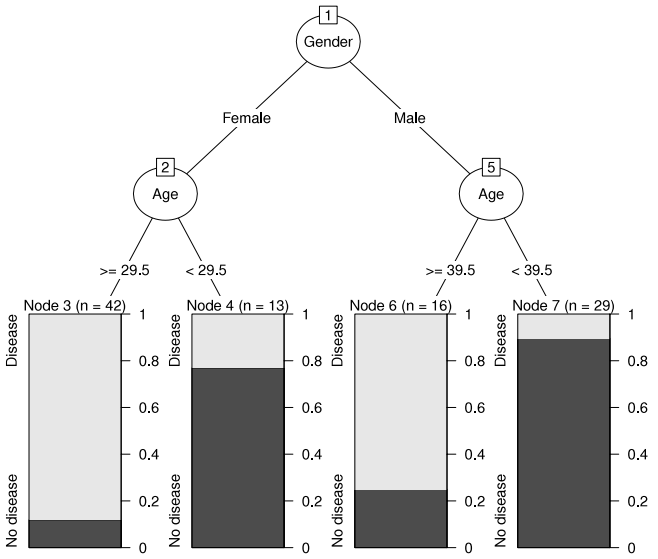


Fig. 1. Example of a classification tree representing an interaction, visualized by `partykit` (Hothorn and Zeileis, 2013). The tree has four end nodes containing subsets of the data. For each subset, the distribution of the data on having a disease is shown. The subsets are more homogeneous with respect to suffering from a disease (i.e., the response variable) than the initial complete dataset.

being categorical or continuous, CART is referred to as classification trees or regression trees respectively (Hastie et al., 2001). The only differences in the algorithm for classification and regression trees concern the criteria for splitting the data and pruning the trees. We refer to Hastie et al. for a detailed discussion.

Let us consider an example of a classification tree. Imagine a dataset with information about whether or not people suffer from a disease (55 people do, 45 people do not), in addition to some background variables (e.g., their gender and age). Fig. 1 shows the classification tree created on these data. This model suggests that there is an interaction between gender and age. In other words, the relation between age and suffering from a disease differs for males and females.

In an attempt to solve the interaction problem in multiple imputation, Burgette and Reiter (2010) used CART for specifying the imputation model in MICE. According to their study, application of a CART imputation engine in MICE can result in "more reliable inferences compared with naive applications of MICE" (p. 2). However, both imputation models resulted in confidence intervals of the parameter estimates that did not cover the corresponding truths. This may partly be due to a lack of incorporation of uncertainty in the imputation models. In addition, it can be explained by imperfect imputation models. It is for example well known that the sequential nature of CART may lead to suboptimal and unstable trees (Hastie et al., 2001; Marshall and Kitsantas, 2012; Strobl et al., 2009). First, for every (sub)set the algorithm seeks the best split, locally optimizing the tree by creating the most homogeneous subsets. The algorithm chooses this best split with no regard for future splits. As a result, the procedure may not produce the best possible tree with the most homogeneous subsets in the leaves. Besides, variable selection is biased in favour of variables with certain characteristics (e.g., variables with many categories), even if these variables are no more informative than their competitors. Lastly, trees can be unstable because of their hierarchical nature where all splits depend on previous splits. This may allow trees created on two datasets that vary only with respect to sample variance to differ markedly from one another.

Random forests differ from CART in that it creates numerous trees, instead of only one. By averaging many trees it reduces the variance and prevalence of unstable trees (Hastie et al., 2001). Variation is produced in the individual trees, resulting in a more robust solution and making the technique more accurate. This variation can be incorporated by, among other procedures, bootstrapping and random input selection (Breiman, 2001). With bootstrapping, before growing each tree, a random selection with replacement is made from the members of the dataset. With random input selection, a small group of input variables is selected for finding the best split, as opposed to using all the variables, as CART does. By the application of random forests in combination with multiple imputation, a degree of uncertainty can be incorporated in the imputation model, making it more eligible for our purpose of creating parameter estimates with appropriate properties. As a result, implementing random forests may be another step forward in solving the interaction problem in imputation.

2.3. Multiple imputation with recursive partitioning

We propose three recursive partitioning techniques to be incorporated as imputation method in the MICE framework: CART, restricted random forests using bootstrapping only (denoted by Forest-boot), and random forests by a combination of bootstrapping and random input selection (denoted by Forest-RI). For the implementation of CART in MICE, suppose a data matrix with multivariate missing values. The missing values are initially imputed by random draws from the observed values on each related variable. Subsequently a tree is fitted on the first variable with at least one missing value, say y_1, using

the remaining variables as predictors. Only members with an observed value on y_1 are taken into account. This results in a tree with several leaves, each containing a subset of the data. A member with a missing value on y_1 is put down this tree and ends up in one of the leaves. From the subset in this leaf one value on y_1 is randomly selected and used for imputation. This procedure is performed for every variable with missing values. A complete cycle along all incomplete variables is repeated several times, yielding one imputed dataset. Ultimately this process is repeated a number of times, yielding multiple imputed datasets. Algorithm 1 shows an implementation of recursive partitioning in MICE. The iterative steps are based on Van Buuren (2012); step 2 is entirely new and describes a general implementation of single tree-recursive partitioning in the MICE framework. In the current study, we used CART to fit a tree (step 2a). However, alternative methods can be plugged in here (e.g., ctree [conditional inference trees], Hothorn et al., 2006; CHAID, Kass, 1980; C4.5, Quinlan, 1993). To apply Algorithm 1 to random forests (i.e., multiple-tree recursive partitioning) some adjustments were needed in step 2. Random forests first draws k bootstrap samples from the complete dataset. One tree is fitted for every bootstrap sample, either with (Forest-RI) or without (Forest-boot) selection of a small group of input variables for finding the best split at each node. Would we have used the averaged tree of this ensemble for imputation, less uncertainty would be incorporated in the imputation model, considering that the averaged tree from a random forest is more optimal and stable than the tree obtained by CART. Since this is undesirable, each and every tree from the ensemble of trees is used separately for imputation in order to represent the uncertainty associated with the missing data. The algorithm for the implementation of random forests in MICE can be found in Appendix A.

To implement CART, Forest-boot and Forest-RI as imputation methods in `mice` (Van Buuren and Groothuis-Oudshoorn, 2011), the `rpart` (Therneau et al., 2013) and `randomForest` (Liaw and Wiener, 2002) packages in R (R Development Core Team, 2013) are used. The R-code for the implementation of CART and random forests in `mice` is given in the supplementary material available in the electronic version of this article (see Appendix C). In the next section, different types of interactions will be discussed, which will be distinguished in testing the imputation methods.

3. Interaction types

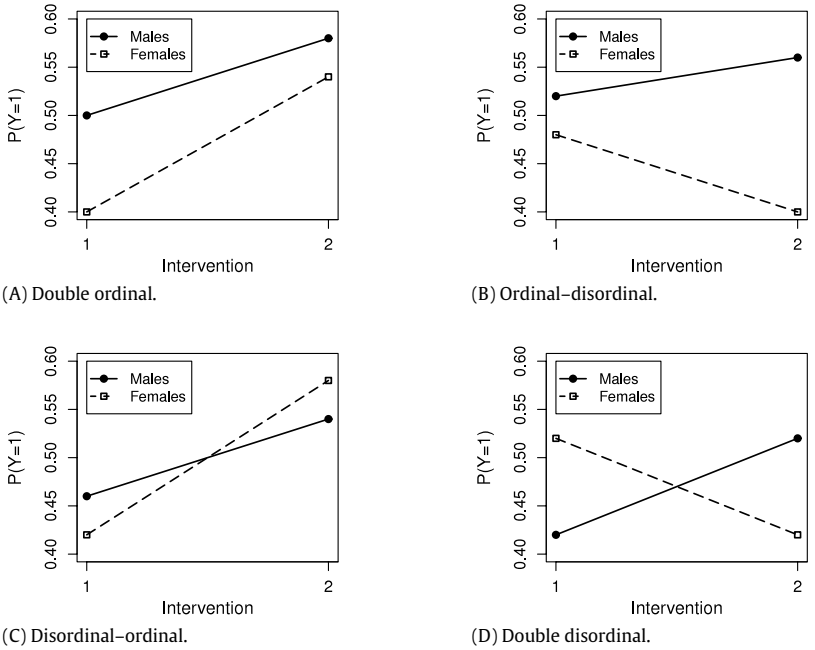
In gaining insight into whether the use of recursive partitioning in multiple imputation is a convenient way to preserve interaction effects, we will distinguish different types of interactions. First, interactions may vary with regard to the correlation (r) between the variables that interact. This correlation may range from -1.0 to 1.0. Second, the effect size of an interaction effect, which implies the strength of the relation between an interaction effect and another (pair of) variable(s), may range. In other words, the relevance of an interaction effect may vary. A third useful distinction is the classification of interactions as being ordinal versus disordinal (Lubin, 1961). These latter two interaction types will be considered regarding categorical variables but one can imagine these to occur with continuous variables too.

In an ordinal situation, the rank order of one variable is constant across the categories of another variable. In contrast, in a disordinal situation the rank order of one variable differs across the categories of the other variable. An interaction may be ordinal with respect to none, one or multiple variables (Aiken and West, 1991). In accordance with Schepers and Van Mechelen (2011) this implies that four types of two-way interactions may be distinguished, and they are presented in Fig. 2. To illustrate, the figure displays four probability profiles of suffering from a disease for males and females across two types of interventions. The rank ordering of the intervention categories is consistent across gender in cases A and C. In other words, with respect to the intervention variable, the interaction is ordinal in these cases. Similarly, the interaction is ordinal with respect to gender in cases A and B, i.e., the rank ordering of gender is constant across the interventions. From a technical point of view, the ordinal–disordinal and disordinal–ordinal interactions are equal. That is, one of the variables in the interaction is ordinal and one of the variables is disordinal with respect to the other.

Algorithm 1 Implementation of recursive partitioning (single tree) in MICE

Suppose a data matrix Y, where Y_j is the jth column of the partially observed variables (ordered to have increasing numbers of missing values so models are build with as much information as possible), p is the number of partially observed variables, Y_j^{obs} is the observed data and Y_j^{mis} is the missing data in the jth column, and $\dot{Y}$ is the currently imputed data matrix Y.

1. For $j = 1, \ldots, p$, fill in initial starting imputations $\dot{Y}_j^0$ by random draws from Y_j^{obs}, and define a data matrix $\dot{Y}$.
2. For $j = 1, \ldots, p$, replace $\dot{Y}_j^0$ as follows, yielding one imputed dataset:
   (a) Fit one tree (using CART or alternative methods) on $\dot{Y}$, restricted to members in Y_j^{obs}. This results in a tree with several leaves, each of which includes a subset of Y_j^{obs}, which will be called donors.
   (b) For members in Y_j^{mis}, determine in which leaf they will end up according to the tree fitted in step 2a.
   (c) For members in Y_j^{mis}, randomly select one Y^{obs} value from the donors of the leaf ended up in step 2b. Replace the originally missing values of $\dot{Y}_j^0$ with these imputation values and append the complete version of $\dot{Y}_j$ to $\dot{Y}$ prior to incrementing j.
3. Repeat step 2 so as to have performed it l (number of iterations) times.
4. Repeat steps 1-3 m times, yielding m imputed sets.



(A) Double ordinal. (B) Ordinal–disordinal.

(C) Disordinal–ordinal. (D) Double disordinal.

Fig. 2. Probability profiles of suffering from a disease for males and females across a set of two interventions. Classifications have been made according to interactions being ordinal or disordinal with respect to the variables that interact.

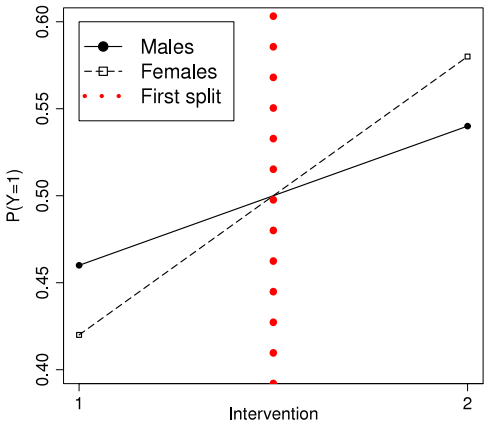


Fig. 3. A disordinal–ordinal interaction. The dotted line indicates a cutpoint for recursive partitioning to start splitting, in order to create homogeneous subsets.

As an illustration of how recursive partitioning handles these interaction types, Fig. 3 displays the disordinal–ordinal interaction. The dotted line indicates a possible cutpoint for recursive partitioning to start splitting, in order to create homogeneous subsets. In our example the dataset is split according to intervention type. That is, one subset is created comprising people in intervention 1 and one subset is created comprising people in intervention 2, where the probability of suffering from a disease is smaller for people in the first compared to the second group. If both subsets are hereafter split according to gender, four subsets would be created and the interaction effect would be detected. A similar procedure can be used for the ordinal–disordinal interaction (case B of Fig. 2). For the double ordinal interaction, either intervention or gender may be selected as first split. Both splits would create more homogeneous subsets compared to the complete data set. After splitting on the remaining predictor variable, the interaction would be detected.

In the presence of a perfectly symmetric double disordinal situation like case D of Fig. 2, there is no obvious split to start with. The two variables show no main effect but a perfect interaction, a situation known as the Exclusive-Or (XOR) problem. Strobl et al. (2009) describe the problem as follows:

> …due to the lack of a marginally detectable main effect, none of the variables may be selected in the first split of a classification tree, and the interaction may never be discovered. …However, a logistic regression model would not be able to identify an effect in any of the variables either, if the interaction was not explicitly included in the logistic regression model. (p. 28)

Recursive partitioning techniques may be able to approximate the XOR problem by two features, namely random fluctuations in the data and random variable selection. In order to start with the first, random fluctuations are present in any data set. These fluctuations may be enhanced by drawing bootstrap samples from the data, as is the case in both the Forest-boot and Forest-RI method. On top of that, by selecting splitting variables randomly (as is the case in Forest-RI), the chance increases that a variable with a weak main effect is selected for splitting. This can be explained by the fact that, at least in some trees, some of the competitors of the variable may not be available for splitting. Consequently, random fluctuations and random variable selection increase the chance that a double disordinal interaction is detected.

The distinguished interaction effects will be observed while considering the use of recursive partitioning in MICE. We will now proceed to the simulation study carried out to investigate which recursive partitioning methods preserve the different interaction effects in multiple imputation. This will be done separately for data with only continuous (Section 4) and only categorical variables (Section 5).

4. Continuous predictor and response variables

4.1. Simulation study

The performance of CART, Forest-boot and Forest-RI were compared with the default imputation method in `mice` for continuous data, i.e., predictive mean matching (denoted by Pmm). The simulation study used to gain insight into the performance of the four imputation methods in the presence of interaction effects can be described on the basis of five components.

Component 1: *Data generation model*. Data were generated using three different regression models, where each model included a two-way interaction effect. The models are specified in Eqs. (4.1)–(4.3), for $i = 1, \ldots, N$ (based on Burgette and Reiter, 2010):

$$y_{1,i} = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \beta_4 x_{8,i} + \beta_5 x_{9,i} + \beta_6 x_{3,i}^2 + \varepsilon_i, \tag{4.1}$$

$$y_{2,i} = \beta_0 + \beta_7 x_{1,i} + \beta_8 x_{2,i} + \beta_9 x_{3,i} + \beta_{10} x_{8,i} + \beta_{11} x_{9,i} + \beta_{12} x_{1,i} x_{2,i} + \varepsilon_i \tag{4.2}$$

and

$$y_{3,i} = \beta_0 + \beta_{13} x_{1,i} + \beta_{14} x_{2,i} + \beta_{15} x_{3,i} + \beta_{16} x_{8,i} + \beta_{17} x_{9,i} + \beta_{18} x_{8,i} x_{9,i} + \varepsilon_i, \tag{4.3}$$

where the intercept $\beta_0 = 0$ and the errors ε_i had independent, standard normal distributions. Artificial data with 10 predictors were randomly drawn from a multivariate normal distribution where all of the first four predictor variables (x_1 to x_4) had pairwise correlations of $r = 0.5$ and all of the last six predictor variables (x_5 to x_{10}) had pairwise correlations of $r = 0.3$. As a result, models (4.1)–(4.3) contained interaction effects where the variables that interact had correlations of $r = 1.0$, $r = 0.5$ and $r = 0.3$ respectively. To clarify, not all predictor variables were part of the model under study and predictor variables 1–4 were not related to predictor variables 5–10.

Component 2: *Design factor*. We varied the values of the effect size (f^2) of the three interaction terms, having three levels: a small effect, a medium effect and a large effect (Cohen, 1988). This corresponds to $f^2 = 0.02$, $f^2 = 0.15$ and $f^2 = 0.35$. We realized these effect sizes by varying the values of parameters β_6, β_{12} and β_{18}, while adapting the other parameters such that the total explained variance of the dependent variable was approximately 50%. The exact values of the parameters can be found in Appendix B.

Component 3: *Missing data creation*. From each of the 3×3 combination of model and effect size, $N = 1000$ observations were simulated. Then, 50% univariate missing data were created in y via a missing at random mechanism that depends on x_9 and x_{10}. For the purpose of improving imputation procedures, the auxiliary variables (i.e., $x_4, x_5, x_6, x_7, x_{10}$) were also included in the missing data model (Collins et al., 2001).

Component 4: *Parameter values that control aspects of the MICE-algorithm or the tree fitting*. Since the missing data were univariate, the number of iterations l (step 3 of Algorithm 1) was set to 1. Steps 1–3 of Algorithm 1 were repeated $m = 20$ times (Graham et al., 2007), resulting in 20 imputed datasets. A minimum leaf size of 5 was used for creating the regression trees, and CART only applied those splits that decreased the overall lack of fit by at least a factor 0.0001. This resulted in relatively large trees which was recommended to minimize bias. The number of bootstrap samples k, taken from the complete dataset in Forest-boot and Forest-RI, was set to 100 (i.e., 100 trees were created) to ensure that every member was used for fitting a tree at least a few times. Besides, 3 input variables were randomly selected for finding the best split at each node in Forest-RI.

Component 5: *Outcome measures*. The performance of the methods was evaluated over 200 simulations on the following outcome variables (based on Van Buuren, 2012): bias, coverage, width of the confidence interval, and estimated proportion of the variance attributable to the missing data ($\hat{\lambda}$). The bias, which is the average difference between the true value of the parameter being estimated (the estimand) and the value of the estimate, should be close to 0. The coverage is the percentage of cases where the value of the estimand is located within the 95% confidence interval around the estimate and should be 95% or higher. The width of the 95% confidence interval should be as small as possible (as long as coverage does not fall below 95%) and is an indicator of statistical efficiency. Lastly, as the proportion of the variance attributable to the missing data, $\hat{\lambda}$ is an indicator of the severity of the missing data problem.

Table 1
Statistical properties of parameter estimates for model (4.1), containing an quadratic relation (considered to be an interaction effect between variables that have a correlation $r = 1.0$) with a medium effect size. β_6 is the parameter of the interaction effect.

β	Method	Bias	Cov[a]	CI[b]	$\hat{\lambda}$[c]	β	Method	Bias	Cov[a]	CI[b]	$\hat{\lambda}$[c]
β_0	Pmm	0.120	0.42	0.223	0.456	β_4	Pmm	0.001	0.97	0.199	0.493
	CART	0.037	0.82	0.180	0.198		CART	−0.031	0.74	0.155	0.200
	Forest-boot	0.047	0.83	0.207	0.345		Forest-boot	−0.053	0.77	0.179	0.357
	Forest-RI	0.067	0.73	0.213	0.355		Forest-RI	−0.062	0.77	0.183	0.364
β_1	Pmm	−0.002	0.95	0.233	0.500	β_5	Pmm	−0.001	0.95	0.225	0.592
	CART	−0.026	0.78	0.181	0.205		CART	−0.058	0.58	0.158	0.235
	Forest-boot	−0.043	0.82	0.206	0.341		Forest-boot	−0.083	0.57	0.187	0.407
	Forest-RI	−0.040	0.89	0.214	0.364		Forest-RI	−0.089	0.57	0.193	0.419
β_2	Pmm	−0.011	0.95	0.229	0.482	β_6	Pmm	−0.109	0.08	0.119	0.353
	CART	−0.044	0.72	0.181	0.204		CART	−0.034	0.69	0.108	0.231
	Forest-boot	−0.056	0.80	0.210	0.365		Forest-boot	−0.046	0.69	0.126	0.389
	Forest-RI	−0.049	0.88	0.214	0.363		Forest-RI	−0.067	0.47	0.129	0.393
β_3	Pmm	0.001	0.93	0.232	0.492						
	CART	0.029	0.81	0.182	0.209						
	Forest-boot	0.018	0.92	0.210	0.359						
	Forest-RI	−0.011	0.96	0.216	0.371						

[a] Coverage, i.e., the percentage of cases where the value of the estimand is located within the 95% confidence interval around the estimate.
[b] Width of the 95% confidence interval around the estimate.
[c] Estimated proportion of the variance attributable to the missing data.

Table 2
Statistical properties of interaction parameter estimates in the context of continuous predictor and response variables.

β	r[a]	Method	Small effect size				Medium effect size				Large effect size			
			Bias	Cov[b]	CI[c]	$\hat{\lambda}$[d]	Bias	Cov[a]	CI[b]	$\hat{\lambda}$[c]	Bias	Cov[b]	CI[c]	$\hat{\lambda}$[d]
β_6	1.0	Pmm	−0.040	0.74	0.11	0.321	−0.109	0.08	0.119	0.353	−0.161	0.01	0.129	0.396
		CART	−0.007	0.82	0.105	0.211	−0.034	0.69	0.108	0.231	−0.042	0.60	0.109	0.258
		Forest-boot	−0.014	0.92	0.123	0.371	−0.046	0.69	0.126	0.389	−0.059	0.51	0.124	0.390
		Forest-RI	−0.020	0.93	0.125	0.373	−0.067	0.47	0.129	0.393	−0.096	0.21	0.131	0.416
β_{12}	0.5	Pmm	−0.040	0.80	0.139	0.325	−0.119	0.14	0.149	0.359	−0.179	0.03	0.164	0.405
		CART	−0.013	0.82	0.133	0.212	−0.057	0.58	0.135	0.223	−0.086	0.37	0.140	0.247
		Forest-boot	−0.018	0.94	0.154	0.361	−0.071	0.59	0.157	0.381	−0.106	0.29	0.162	0.399
		Forest-RI	−0.026	0.92	0.158	0.374	−0.090	0.43	0.160	0.381	−0.139	0.08	0.168	0.417
β_{18}	0.3	Pmm	−0.077	0.49	0.154	0.371	−0.227	0	0.161	0.385	−0.346	0	0.176	0.415
		CART	−0.049	0.71	0.147	0.253	−0.154	0.07	0.151	0.271	−0.230	0.01	0.158	0.285
		Forest-boot	−0.057	0.79	0.173	0.416	−0.170	0.02	0.179	0.439	−0.254	0	0.188	0.458
		Forest-RI	−0.061	0.82	0.177	0.428	−0.185	0	0.183	0.448	−0.282	0	0.192	0.465

[a] Correlation between the variables that interact.
[b] Coverage, i.e., the percentage of cases where the value of the estimand is located within the 95% confidence interval around the estimate.
[c] Width of the 95% confidence interval around the estimate.
[d] Estimated proportion of the variance attributable to the missing data.

4.2. Results

The results for model (4.1), containing a medium effect size interaction, are presented in Table 1. The problem noted in the introduction is clearly illustrated. That is, the default application of `mice` (i.e., Pmm) does not automatically incorporate possible interaction effects, leading to biased estimates of the interaction parameter (−0.109) and a low coverage (0.08) after imputation. When we focus on the parameter estimates of the interaction effect over the four imputation methods, it turns out that recursive partitioning performs properly better than Pmm. However, for the recursive partitioning imputation methods, the biases of the main effects are somewhat larger and the coverages are smaller compared to Pmm. This holds in particular for β_4 and β_5, which are the parameter estimates for the main effects of predictor variables that have a correlation of $r = 0.3$ with other predictors.

As the focus of the paper is on the statistical properties of interaction parameter estimates, for the remainder designs of models (4.1)–(4.3), we present only the results with respect to the interaction effects. Concerning the main effects of these designs, similar patterns were obtained as shown in Table 1. That is, results somewhat deteriorate by using recursive partitioning instead of Pmm for imputation, especially for main effects of predictor variables that have a relatively small correlation with other predictors. Table 2 presents the results of the simulation study regarding the interaction parameters β_6, β_{12} and β_{18}, with respect to the three investigated effect sizes. The statistical properties of estimates of these parameters will be discussed considering the imputation methods, the effect sizes of the interaction effects and the correlation between variables that interact.

All four imputation methods underestimate the interaction effects to some degree, but they show that there are substantial gains possible in using the three recursive partitioning methods rather than standard `mice` for imputation. Firstly, the

estimates are closer to the true values when recursive partitioning is used as imputation method, i.e., the biases are smaller. Secondly, Table 2 shows that CART, by all means, is more efficient than Pmm. That is, confidence intervals are smaller but not at the expense of coverages. Thirdly, the advantage of random forests in incorporating uncertainty in the imputation model is shown by the wider confidence intervals of Forest-boot and Forest-RI, with little difference between them. These wider confidence intervals result in higher coverages for the design with small effect sizes, despite of the somewhat higher biases compared to CART. In contrast, the wider confidence intervals for Forest-boot and Forest-RI cannot compete with the larger biases in the designs with medium and large effect sizes, which results in lower coverages. The $\hat{\lambda}$ parameters estimated are around 0.2 when CART is used for imputation, indicating a modest missing data problem. The higher values for $\hat{\lambda}$ when random forests and especially Pmm are used indicate a more difficult problem for these imputation methods, in which the final statistical inferences are more highly dependent on the way in which the missing data were handled (Van Buuren, 2012).

When it comes to the effect sizes it may further be noted that, in general, though none of the recursive partitioning methods is far off when the effect size of an interaction is small, results deteriorate as the effect size of an interaction increases. Also with regard to the three interaction effects with varying correlations between the variables that interact, clear graduations can be seen. Results improve as the correlation between variables that interact increases, namely biases are smaller and the imputation methods are more efficient (i.e., higher coverages while confidence intervals are less wide). The tendencies regarding effect sizes of interactions and correlations between variables, can combine to result in coverages that are extremely low. In particular when interaction effects between variables that have a relatively low correlation, have a high effect size.

We conclude with a note on the computational performance of the four imputation methods. Running the simulations for each 3×3 combination of model and effect size on an Intel Core i7 took 2.8 min for pmm, 3.7 min for CART, 80 min for Forest-boot and 72.1 min for the Forest-RI imputation approach (with 200 simulations). Both model and effect size had no appreciable effect on the computation time.

5. Categorical predictor and response variables

In this section, the performance of CART, Forest-boot and Forest-RI as imputation method in `mice` is investigated regarding categorical predictor and response variables.

5.1. Simulation study

The performance of the three recursive partitioning imputation methods will be compared with the following standard application of `mice`: logistic regression imputation (denoted by Logreg). In the performed simulation study, the same five components can be recognized as in the simulation study described in Section 4.1.

Component 1: *Data generation model*. Data were generated using three different logistic regression models: one with a double ordinal interaction, one with a disordinal–ordinal interaction and one with a double disordinal interaction. The models are specified in Eqs. (5.1)–(5.3) respectively:

$$\text{logit}[P(Y_1 = 1)] = \alpha_0 + \alpha_1 d_1 + \alpha_2 d_2 + \alpha_3 d_3 + \alpha_4 d_4 + \alpha_5 d_8 + \alpha_6 d_9 + \alpha_7 d_1 d_2, \tag{5.1}$$

$$\text{logit}[P(Y_2 = 1)] = \alpha_0 + \alpha_8 d_1 + \alpha_9 d_2 + \alpha_{10} d_3 + \alpha_{11} d_4 + \alpha_{12} d_8 + \alpha_{13} d_9 + \alpha_{14} d_3 d_4 \tag{5.2}$$

and

$$\text{logit}[P(Y_3 = 1)] = \alpha_0 + \alpha_{15} d_1 + \alpha_{16} d_2 + \alpha_{17} d_3 + \alpha_{18} d_4 + \alpha_{19} d_8 + \alpha_{20} d_9 + \alpha_{21} d_8 d_9, \tag{5.3}$$

where the intercept $\alpha_0 = 0$, and d_1, d_2, d_3, d_4, d_8 and d_9 are binary predictor variables (i.e., dummies). Artificial data including 10 predictor variables were randomly drawn from a binomial distribution. To clarify, the predictor variables were uncorrelated and they were not all part of the models under study. Variables that were part of the models consisted of two categories, the other variables had three categories. Incorporation of the interaction types was realized by the values assigned to the parameters α that were part of the interaction effects (e.g., α_1, α_2, α_7 in model (5.1)).

Component 2: *Design factor*. We varied the values of the effect size of the three interaction terms, using the odds ratio as an index. Haddock et al. (1998) have provided guidelines for interpreting the magnitude of an odds ratio: "As general rules of thumb, odds ratios close to 1.0 represent a weak relationship between variables, whereas odds ratios over 3.0 for positive associations (less than one-third for negative associations) indicate strong relationships" (p. 342). We realized weak and strong effect sizes by varying the values of parameters α_7, α_{14} and α_{21}, while adapting the other parameters such that the base rate was approximately 0.50 for all models. The exact values of the parameters can be found in Appendix B.

Component 3: *Missing data creation*. For each 3×2 combination of model and effect size, 1000 observations were simulated. Then, 50% univariate missing data were created in Y via a missing at random mechanism that depends on d_9 and d_{10}. Variables not part of the model under study (i.e., d_5, d_6, d_7, d_{10}) were also included in the missing data model.

Component 4: *Parameter values that control aspects of the MICE-algorithm or the tree fitting*. The number of iterations, the number of imputed datasets, and the values of parameters that control aspects of the tree fitting were equal to the ones used in the simulation with continuous predictor and response variables. They can be found in Section 4.1.

Table 3
Statistical properties of parameter estimates for model (5.1), containing a double ordinal interaction with large effect size. α_7 is the parameter of the interaction effect.

α	Method	Bias	Cov[a]	CI[b]	$\hat{\lambda}$[c]	α	Method	Bias	Cov[a]	CI[b]	$\hat{\lambda}$[c]
α_0	Logreg	−0.185	0.90	1.146	0.479	α_4	Logreg	0.005	0.96	0.853	0.498
	CART	−0.040	0.87	0.954	0.296		CART	0.056	0.85	0.706	0.303
	Forest-boot	−0.084	0.87	0.985	0.325		Forest-boot	0.047	0.86	0.723	0.319
	Forest-RI	−0.095	0.89	0.983	0.339		Forest-RI	0.114	0.87	0.721	0.343
α_1	Logreg	0.266	0.88	1.103	0.430	α_5	Logreg	0.015	0.96	0.854	0.500
	CART	−0.091	0.90	0.963	0.297		CART	0.084	0.82	0.711	0.311
	Forest-boot	−0.082	0.92	0.989	0.321		Forest-boot	0.067	0.83	0.728	0.328
	Forest-RI	−0.055	0.94	0.994	0.344		Forest-RI	0.134	0.83	0.721	0.343
α_2	Logreg	0.269	0.87	1.096	0.421	α_6	Logreg	0.036	0.95	0.838	0.496
	CART	−0.054	0.88	0.952	0.296		CART	0.126	0.87	0.693	0.299
	Forest-boot	0.008	0.87	0.981	0.324		Forest-boot	0.113	0.92	0.711	0.318
	Forest-RI	−0.013	0.92	0.988	0.347		Forest-RI	0.135	0.91	0.708	0.336
α_3	Logreg	−0.005	0.95	0.847	0.505	α_7	Logreg	−0.533	0.80	1.460	0.346
	CART	−0.102	0.88	0.692	0.294		CART	0.091	0.92	1.411	0.300
	Forest-boot	−0.085	0.90	0.711	0.319		Forest-boot	0.161	0.90	1.454	0.321
	Forest-RI	−0.110	0.91	0.711	0.340		Forest-RI	−0.075	0.97	1.453	0.350

[a] Coverage, i.e., the percentage of cases where the value of the estimand is located within the 95% confidence interval around the estimate.
[b] Width of the 95% confidence interval around the estimate.
[c] Estimated proportion of the variance attributable to the missing data.

Table 4
Statistical properties of interaction parameter estimates in the context of categorical predictor and response variables.

α	Interaction type	Method	Small effect size				Large effect size			
			Bias	Cov[a]	CI[b]	$\hat{\lambda}$[c]	Bias	Cov[a]	CI[b]	$\hat{\lambda}$[c]
α_7	Double ordinal	Logreg	−0.211	0.99	1.385	0.336	−0.533	0.80	1.460	0.346
		CART	−0.016	0.84	1.329	0.289	0.091	0.92	1.411	0.300
		Forest-boot	−0.028	0.89	1.374	0.325	0.161	0.90	1.454	0.321
		Forest-RI	−0.083	0.93	1.374	0.342	−0.075	0.97	1.453	0.350
α_{14}	Disordinal–ordinal	Logreg	−0.218	0.99	1.356	0.336	−0.542	0.77	1.403	0.333
		CART	−0.138	0.94	1.303	0.295	−0.181	0.86	1.367	0.304
		Forest-boot	−0.132	0.97	1.345	0.330	−0.181	0.92	1.401	0.327
		Forest-RI	−0.156	0.97	1.331	0.331	−0.187	0.91	1.397	0.344
α_{21}	Double disordinal	Logreg	−0.208	0.99	1.308	0.328	−0.546	0.71	1.336	0.330
		CART	−0.166	0.98	1.264	0.289	−0.313	0.81	1.301	0.298
		Forest-boot	−0.168	0.99	1.307	0.325	−0.324	0.87	1.333	0.319
		Forest-RI	−0.168	0.99	1.310	0.338	−0.366	0.86	1.331	0.331

[a] Coverage, i.e., the percentage of cases where the value of the estimand is located within the 95% confidence interval around the estimate.
[b] Width of the 95% confidence interval around the estimate.
[c] Estimated proportion of the variance attributable to the missing data.

Component 5: *Outcome measures*. The performance of the methods was evaluated over 200 simulations with the same outcome variables as described in Section 4.1: bias, coverage, width of the confidence interval, and the estimated proportion of the variance attributable to the missing data ($\hat{\lambda}$).

5.2. Results

The results for model (5.1) containing a double ordinal interaction with large effect size are presented in Table 3. The interaction problem is again clearly illustrated. That is, the parameters related to the double ordinal interaction (i.e., $\alpha_1, \alpha_2, \alpha_7$) are seriously biased when the default application of `mice` (i.e., Logreg) is used for imputation. When recursive partitioning techniques are used for imputation, these parameter estimates are severely less biased. The smaller biases result in larger coverages, despite the smaller confidence intervals for recursive partitioning techniques. With regard to the parameters not part of the interaction effect, biases have become somewhat larger and coverages have become lower as recursive partitioning is used for imputation. Similar patterns are found for the models with a disordinal–ordinal and double disordinal interaction effect. In essence, results improve for parameters part of the interaction while the results for the remaining parameters slightly deteriorate. Since these patterns are similar over designs, we will just focus on the statistical properties of the interaction parameter estimates (i.e., $\alpha_7, \alpha_{14}, \alpha_{21}$) for the remaining designs. These results are presented in Table 4 and will be examined in more detail concerning the four imputation methods, the effect sizes of the interaction effects and the interaction types.

First, in general, all four imputation methods underestimate the interaction effects, but Logreg does this to the greatest extent. The recursive partitioning methods perform considerably better when it comes to correctly estimating the interaction

parameters. Besides, it is notable that using recursive partitioning for imputation is generally more efficient than Logreg. This is indicated by the smaller confidence intervals while coverages are higher. Comparing the three recursive partitioning methods shows that random forests entail a small amount of extra uncertainty in the imputation models compared to CART, i.e., the confidence intervals are somewhat wider. Hereby the random selection of splitting variables in Forest-RI was not of added value. Resulting from the wider confidence intervals for random forests, the coverages also increase as this technique is used for imputation. Lastly, the estimated $\hat{\lambda}$ parameters are all around 0.3, which implies that the methods deal with moderately large fractions of missing information. The values of $\hat{\lambda}$ do not diverge much, though the problem of missing data shows least severe as CART is used for imputation.

With respect to the importance of the interaction effects, none of the recursive partitioning methods seem way off when these are small. Yet again, the results for the interactions with small effect sizes are generally better (i.e., smaller biases and higher coverages) than the results for the interactions with larger effect sizes. Lastly, with regard to the interaction types it turns out that recursive partitioning methods have most difficulty with correctly estimating a double disordinal interaction. Parameter estimates for the interaction effects were least biased when the interaction was double ordinal.

We conclude again with a note on the computational performance of the default application in `mice` and the recursive partitioning methods. On an Intel Core i7, it took 1.4 min for Logreg, 3.1 min for CART, 210 min for Forest-boot and 208 min for the Forest-RI imputation approach to run the simulations for each 3×2 combination of model and effect size (with 200 simulations). These computation times are stable across the models and effect sizes varied in the simulation study.

6. Discussion

We implemented three recursive partitioning techniques that incorporate interaction effects in the data, as imputation method in `mice`: CART, restricted random forests using bootstrapping only and random forests by a combination of bootstrapping and random input selection. We studied the bias and coverage of parameter estimates after imputation by these methods. In doing this, we replicated and extended the study of Burgette and Reiter (2010), who examined the performance of CART in MICE for continuous variables. They conclude that CART as imputation method can result in more reliable inferences compared with standard applications of MICE based on main-effects generalized linear models. We obtained similar results for the case studied by Burgette and Reiter, using predictive mean matching as standard application. We also examined the application of MICE based on Bayesian linear regression analyses but this method performed worse. As extension to the study of Burgette and Reiter, we investigated the use of random forests, data with categorical predictor and response variables, various effect sizes of interaction effects, the correlation structure of the data and the type of interactions.

Our main result is that, regardless of variables being continuous or categorical, CART preserves interaction effects best (i.e., not only compared to standard applications but also compared to random forest). Also, larger interaction effects are more difficult to impute. Furthermore, the quality of parameter estimates deteriorates as the correlation between variables that interact decreases. Lastly, double ordinal interactions prove to be easiest to preserve automatically. These four results are discussed in more detail in the following paragraphs.

Both CART and random forests are conservative in the sense that if bias occurs, it is towards zero, but we found CART to create the least biased parameter estimates. The imperfect imputation models that led to the bias of both techniques may have emerged from the presence of main effects in the data. That is, recursive partitioning techniques have difficulty in modelling linear main effects. The main effects are hard to capture because, due to the binary tree model, "it would take many fortuitous splits to recreate the structure" (Hastie et al., 2001, p. 313). We expect that this problem will also occur using other recursive partitioning techniques. A possible solution to this problem has been offered by STIMA (Dusseldorp et al., 2010), which combines a linear main effects model with recursive partitioning. The difficulty with modelling linear main effects also explains why in our study the recursive partitioning imputation methods led to biases that are somewhat higher for the main effects compared to standard applications of MICE. The higher biases for the interaction effects by random forests compared to CART may be explained by interactions that are missed in the tree building process due to drawing bootstrap samples and the (low) number of randomly preselected variables (Strobl et al., 2009). We therefore conclude that CART preserves interaction effect best, even though random forests did account for somewhat more uncertainty associated with the missing data. More generally, we conclude that recursive partitioning methods are recommended over standard applications of MICE if one has presumptions of interaction effects, as the gain in preserving interaction effects outweighs the somewhat higher biases for the main effects.

Higher effect sizes of interaction effects were found to be associated with larger biases and lower coverages, where the biases we reported were defined as the absolute difference between the estimand and the estimate. However, the relative bias (i.e., absolute bias divided by the value of the estimand) turned out to be constant or even somewhat decreasing over effect sizes. These results can be explained as follows. The imputation methods seem to pull estimates to zero, i.e., they are conservative. In absolute terms, the impact of this bias is larger as effect sizes are higher. The confidence intervals became also larger as effect sizes increased, but only to a limited extent. This combination of limited enlargement of the confidence intervals and biases that increased, led to coverages that were poor as effect sizes were high.

The statistical quality of interaction parameters improved as the correlation between variables that interact increased. To illustrate, Fig. 4 presents the probability of being selected as first splitting variable per predictor of model (4.3). This probability is high for predictors that have a mutual correlation of $r = 0.5$ (i.e., x_1, x_2, x_3) compared with variables that have a mutual correlation of $r = 0.3$ (i.e., x_8, x_9). The explanation for this finding is that a higher correlation between predictors

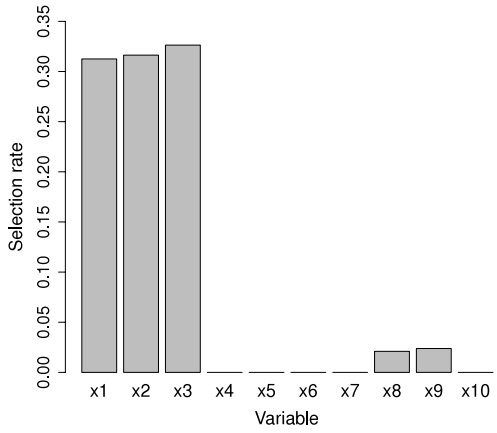


Fig. 4. Relative selection rates for the first split regarding model (4.3), containing an interaction between x_8 and x_9 with a large effect size. Predictors x_1, x_2 and x_3 have mutual correlations of $r = 0.5$, and predictors x_8 and x_9 have mutual correlations of $r = 0.3$. Remaining variables are not related to the response variable. Parameter values for the predictors are equal.

implies a higher correlation between each predictor with the response variable. As a consequence, recursive partitioning techniques prefer these highly correlated variables for splitting over variables that are less predictive of the response variable. This also explains why interaction effects between variables that have a low correlation are harder to detect.

With regard to interactions being ordinal or disordinal, we found that a double ordinal interaction is best preserved by the imputation methods. This might be due to the double main effect that is part of such an interaction, which makes it easier to detect. In line with the theory it appeared most difficult for recursive partitioning imputation methods to preserve a double disordinal interaction effect. Nevertheless, specific caution for imputing data with a possible double disordinal interaction is not in place. Problems were expected to arise in the presence of a perfectly symmetric double disordinal interaction, which is an unrealistic situation in real data. Beyond that, it appears that in practice the theory that two interaction effects cancel out their main effect, is not problematic.

With certainty, there is still room for improvement of recursive partitioning as imputation method in order for it to work in each and every situation. We concentrated on the most popular recursive partitioning methods, i.e., CART and random forests. As a suggestion, one could consider alternative methods like conditional inference trees (a framework in which tree-structured models are embedded into a theory of conditional inference procedures; Hothorn et al., 2006), CHAID (Kass, 1980), C4.5 (Quinlan, 1993), MARS (a non-parametric regression technique that automatically models nonlinearities and interactions; Friedman, 1991), GUIDE (a machine learning algorithm for generalized, unbiased, interaction detection and estimation; Loh, 2002) and STIMA (Dusseldorp et al., 2010). Though we considered various designs, our study is still limited in terms of results that may not be generalizable due to our particular choices of data generation. As a second suggestion for future research, we therefore propose an extension of the simulation study by systematically varying the structure of the data, the missing data pattern (i.e., univariate, multivariate) and the missing data mechanism (i.e., missing at random, missing not at random). Lastly, a general remark can be made towards our goal to have imputation models that fit to the data in an automatic fashion: automation of imputation methods gives no license to stop thinking about structures of the data and missingness.

Beyond the fact that there is still room for improvement, it can be concluded that recursive partitioning techniques are valuable for imputing datasets containing interaction effects. We have shown that, compared with standard applications, substantial gains are possible in using recursive partitioning as imputation method in multiple imputation, when interaction effects are present in the data. The recursive partitioning imputation methods we presented allow, to a greater or lesser extent, for imputation of missing values while automatically accounting for interaction effects in the data at hand and the uncertainty associated with the missing data. So far, no imputation methods are available that meet this requirement.

Appendix A. Implementation of random forests in MICE

Algorithm for the implementation of Forest-boot and Forest-RI in MICE. This algorithm can be generalized to random forests techniques other than Forest-boot and Forest-RI (e.g., random forests based on conditional inference trees; Hothorn et al., 2006), by adjusting the method used to fit the k trees in step 2b.

Appendix B. Parameter values used in the simulation study

Values of the parameters weights β (Table B.1) and α (Table B.2) to generate the data. The data with continuous predictor and response variables are generated from the models described in Section 4.1, and the data with categorical predictor and response variables are generated from the models in Section 5.1. The values of f^2 are the effect sizes of the interaction terms. The values of the interaction parameters are in boldface.

Table B.1
True parameters values from models (4.1)–(4.3).

Correlation		$r = 1.0$				$r = 0.5$				$r = 0.3$		
f^2		0.020	0.15	0.35		0.02	0.15	0.35		0.02	0.15	0.35
	β_1	0.34	0.31	0.28	β_7	0.34	0.31	0.28	β_{13}	0.34	0.31	0.28
	β_2	0.34	0.31	0.28	β_8	0.34	0.31	0.28	β_{14}	0.34	0.31	0.28
	β_3	0.34	0.31	0.28	β_9	0.34	0.31	0.28	β_{15}	0.34	0.31	0.28
	β_4	0.34	0.31	0.28	β_{10}	0.34	0.31	0.28	β_{16}	0.34	0.31	0.28
	β_5	0.34	0.31	0.28	β_{11}	0.34	0.31	0.28	β_{17}	0.34	0.31	0.28
	β_6	**0.11**	**0.28**	**0.42**	β_{12}	**0.13**	**0.35**	**0.53**	β_{18}	**0.13**	**0.37**	**0.57**

Table B.2
True parameters values from models (5.1)–(5.3).

Type		Double ordinal			Disordinal–ordinal			Double disordinal	
Effect size		Small	High		Small	High		Small	High
	α_1	0.5	0.5	α_8	0.7	0.7	α_{15}	0.4	0.4
	α_2	1	1	α_9	0.4	0.5	α_{16}	0.5	0.4
	α_3	0.4	0.5	α_{11}	0.7	0.7	α_{17}	0.4	0.4
	α_4	−0.5	−1	α_{11}	−0.2	−0.2	α_{18}	−1.1	−1.3
	α_5	−1.1	−1	α_{12}	−1.1	−1.1	α_{19}	−0.2	−0.2
	α_6	−0.5	−0.5	α_{13}	−0.7	−1.1	α_{20}	−0.2	−0.2
	α_7	**0.4**	**1.1**	α_{14}	**0.4**	**1.1**	α_{21}	**0.4**	**1.1**

Algorithm A.1 Implementation of random forests in MICE

Suppose a data matrix Y, where Y_j is the jth column of the partially observed variables (ordered to have increasing numbers of missing values so models are build with as much information as possible), p is the number of partially observed variables, Y_j^{obs} is the observed data and Y_j^{mis} is the missing data in the jth column, and $\dot{Y}$ is the currently imputed data matrix Y.

1. For $j = 1, \ldots, p$, fill in initial starting imputations $\dot{Y}_j^0$ by random draws from Y_j^{obs}, and define a data matrix $\dot{Y}$.
2. For $j = 1, \ldots, p$, replace $\dot{Y}_j^0$ as follows, yielding one imputed dataset:
   (a) Draw k bootstrap samples from $\dot{Y}$, restricted to members in Y_j^{obs}
   (b) Fit one tree on every bootstrap sample drawn in step 2a, either with (Forest-RI) or without (Forest-boot) selection of a small group of input variables for finding the best split at each node. This results in k trees, where every tree has several leaves. Each leaf includes a subset of Y_j^{obs}, which will be called donors.
   (c) For members in Y_j^{mis}, determine in which leaf they will end up according to the k trees fitted in step 2b. This results in k leafs with donors per member of Y_j^{mis}.
   (d) For members in Y_j^{mis}, take all donors from the k leafs ended up in step 2c together and randomly select one Y^{obs} value from the donors. Replace the originally missing values of $\dot{Y}_j^0$ with these imputation values and append the complete version of $\dot{Y}_j$ to $\dot{Y}$ prior to incrementing j.
3. Repeat step 2 so as to have performed it l (number of iterations) times.
4. Repeat steps 1–3 m times, yielding m imputed sets.

Appendix C. Supplementary data

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.csda.2013.10.025.

References

Aiken, L.S., West, S.G., 1991. Multiple Regression: Testing and Interpreting Interactions. Sage, Newbury Park, CA.
Breiman, L., 2001. Random forest. Mach. Learn. 45, 5–32.
Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J., 1984. Classification and Regression Trees. Wadsworth & Brooks/Cole Advanced Books & Software, Monterey, CA.
Burgette, L.F., Reiter, K.P., 2010. Multiple imputation for missing data via sequential regression trees. Amer. J. Epidemiol. 172, 1070–1076.
Cohen, J., 1988. Statistical Power Analysis for the Behavioral Sciences. Lawrence Erlbaum Associates, Hillsdale, New Jersey.
Collins, L.M., Schafer, J.L., Kam, C.M., 2001. A comparison of inclusive and restrictive strategies in modern missing data procedures. Psychol. Methods 6, 330–351.
Dusseldorp, E., Conversano, C., Van Os, B.J., 2010. Combining an additive and tree-based regression model simultaneously: STIMA. J. Comput. Graph. Statist. 19, 514–530.
Friedman, J.H., 1991. Multivariate adaptive regression splines. Ann. Statist. 19, 1–67.
Graham, J.W., Olchowski, A.E., Gilreath, T.D., 2007. How many imputations are really needed? Some practical clarifications of multiple imputation theory. Prevention Sci. 8, 206–213.

Haddock, C., Rindskopf, D., Shadish, W., 1998. Using odds ratios as effect sizes for meta-analysis of dichotomous data: a primeur on methods and issues. Psychol. Methods 3, 339–353.
Hand, D.J., 1997. Construction and Assessment of Classification Rules. Wiley, Chichester.
Hastie, T., Tibshirani, R., Friedman, J., 2001. The Elements of Statistical Learning; Data Mining, Inference, and Prediction, second ed. Springer Verlag, New York.
Hothorn, T., Hornik, K., Zeileis, A., 2006. Unbiased recursive partitioning: a conditional inference framework. J. Comput. Graph. Statist. 15, 651–674.
Hothorn, T., Zeileis, A., 2013. Partykit: a toolkit for recursive partitioning. R package version 0.1-6. URL: http://CRAN.R-project.org/package=partykit.
Iacus, S.M., Porro, G., 2007. Missing data imputation, matching and other applications of random recursive partitioning. Comput. Statist. Data Anal. 52, 773–789.
Iacus, S.M., Porro, G., 2008. Invariant and metric free proximities for data matching: an R package. J. Stat. Softw. 25, 1–22.
Kass, G.V., 1980. An exploratory technique for investigating large quantities of categorical data. J. R. Stat. Soc. 29, 119–127.
Liaw, A., Wiener, M., 2002. Classification and regression by randomforest. R News 2, 18–22. URL: http://CRAN.R-project.org/package=randomForest.
Loh, W.Y., 2002. Regression trees with unbiased variable selection and interaction detection. Statist. Sinica 12, 361–386.
Lubin, A., 1961. The interpretation of significant interaction. Educ. Psychol. Meas. 21, 807–817.
Marshall, R.J., Kitsantas, P., 2012. Stability and structure of cart and span search generated data partitions for the analysis of low birth weight. J. Data Sci. 10, 61–73.
Merkle, E.C., Schaffer, V.A., 2011. Binary recursive partitioning: background, methods, and application to psychology. British J. Math. Statist. Psych. 64, 161–181.
Morgan, J.N., Sonquist, J.A., 1963. Problems in the analysis of survey data, and a proposal. J. Amer. Statist. Assoc. 58, 415–434.
Nonyane, B.A.S., Foulkes, A.S., 2007. Multiple imputation and random forests (mirf) for unobservable, high dimensional data. Int. J. Biostat. 3. Article 12.
Quinlan, J.R., 1993. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers.
R Development Core Team. 2013. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. URL: http://www.R-project.org/.
Rubin, D.B., 1996. Multiple imputation after 18+ years. J. Am. Stat. Assoc. 91, 473–489.
Schepers, J., Van Mechelen, I., 2011. A two-mode clustering method to capture the nature of the dominant interaction pattern in large profile data matrices. Psychol. Methods 16, 361–371.
Stekhoven, D.J., Bühlmann, P., 2012. Missforest—non-parametric missing value imputation for mixed-type data. Bioinformatics 1, 112–118.
Strobl, C., Malley, J., Zeileis, A., 2009. An introduction to recursive partitioning: rationale, application and characteristics of classification and regression trees, bagging and random forests. Psychol. Methods 14, 323–348.
Therneau, T., Atkinson, B., Ripley, B., 2013. rpart: recursive partitioning. R package version 4.1-3. URL: http://CRAN.R-project.org/package=rpart.
Van Buuren, S., 2007. Multiple imputation of discrete and continuous data by fully conditional specification. Stat. Methods Med. Res. 16, 219–242.
Van Buuren, S., 2012. Flexible Imputation of Missing Data. Chapman & Hall/CRC, Boca Raton, FL.
Van Buuren, S., Groothuis-Oudshoorn, K., 2011. MICE: multivariate imputation by chained equations in R. J. Stat. Softw. 45, 1–67. URL: http://CRAN.R-project.org/package=mice.