Journal of Data Science 18 (4), 606–631
October 2020
DOI: 10.6339/JDS.202010_18(4).0002

# Tree-Based Missing Value Imputation Using Feature Selection

Heizel Rosado-Galindo*[1] and Saylisse Dávila-Padilla[2]

[1]*Department of Bioengineering, University of Puerto Rico, Mayagüez, Puerto Rico*
[2]*Department of Industrial Engineering, University of Puerto Rico, Mayagüez, Puerto Rico*

**Abstract**

Researchers and practitioners of many areas of knowledge frequently struggle with missing data. Missing data is a problem because almost all standard statistical methods assume that the information is complete. Consequently, missing value imputation offers a solution to this problem. The main contribution of this paper lies on the development of a random forest-based imputation method (TI-FS) that can handle any type of data, including high-dimensional data with non-linear complex interactions. The premise behind the proposed scheme is that a variable can be imputed considering only those variables that are related to it using feature selection. This work compares the performance of the proposed scheme with other two imputation methods commonly used in literature: KNN and missForest. The results suggest that the proposed method can be useful in complex scenarios with categorical variables and a high volume of missing values, while reducing the amount of variables used and their corresponding preliminary imputations.

**Keywords** *K nearest neighbor; missForest; missing data; random forest*

## 1 Introduction

Researchers and practitioners in many areas of knowledge frequently struggle with missing data. Missing data arises in almost all statistical analyses for reasons, such as data collection problems, equipment failures, errors in manual data entry or in cases of non-response items in survey studies with persons (Rogier et al., 2006; Gelman and Hill, 2006). Work performed by Wood et al. (2004) reveals that 89% of the clinical experiments in leading medical journals exhibit missing values. Unfortunately, missing data is a problem because almost all standard statistical methods assume that the information is complete. As a result, the analysis of the data gets complicated, efficiency is lost, statistical power decreases, and parameter estimates may be biased due to the differences between the complete and missing data (Kaiser, 2014).

Researchers often appeal to ad hoc methods, such as case deletion or missing value imputation, to force an incomplete data set into a data set with no missing values (Schafer, 1997). The consequence of case deletion is that potentially valuable data is discarded, which is usually worse than having missing values. *Missing value imputation* (MVI), on the other hand, refers to replacing the missing data with acceptable values, by using the data in the recorded variables (Andridge and Little, 2010). The approach to MVI schemes is typically straightforward. You either delete all instances with at least one missing value; or you generate preliminary imputations for all independent variables, except for the one whose missing value you want to impute. The same missing value is imputed multiple times until it no longer changes as stipulated by a convergence criteria, which can be in terms of the number of iterations or in terms of the change in the value of the missing value.

*Corresponding author. Email: heizel.rosado@upr.edu.


Received November, 2018; Accepted June, 2020

When the missing cases are a small part of the data set (e.g. 5% or less) case deletion could be a reasonable solution to the missing data problem. But, when dealing with high number of missing data, discarding them will lead to losing large amounts of information; without mentioning that the data collection process often requires large amounts of time and money. This is the case when conducting studies that involve clinical trials, for instance, a new cancer treatment where trials require a hefty monetary investment and are only conducted after getting the approval from the health regulatory agency (Sertkaya et al., 2014). On top of that, it will also take, on average, five years to collect the necessary data to perform robust analyses. Many things can lead to missing data during the process (e.g. patients dropping out from the study, problems with data collection) and, thus, knowing the substantial amount of resources it takes to collect it, discarding cases is typically the least attractive option. This is why MVI is a growing area of research, specially among researchers working on experiments that involve high-dimensional data sets.

Literature on mixed-type data imputation is somewhat limited. Most imputation methods are restricted to only one type of variable. For example, stochastic regression imputation (SRI), is used for categorical data exclusively (Sulis and Porcu, 2008), whereas regression imputation, is only used on continuous data. The options fall even shorter when complex mixed-type data comes into play. The first attempt to overcome this gap involved maximum likelihood estimation, combining a multivariate normal model with a Poisson/multinomial model to impute continuous and categorical variables, respectively (Little and Schluchter, 1985). During the last decade, other methods based on decision trees (Stekhoven and Bülmann, 2012) and nearest neighbors (Kowarik and Templ, 2016) have been proposed. Yet, there still a need for new and enhanced techniques that can satisfy the ongoing necessities of big data.

This paper describes TI-FS (tree-based missing value imputation using feature selection), a new imputation method based on random forests (Breiman, 2001), that exploits the relationships among variables by means of feature selection. The premise behind the proposed scheme is that a variable can be imputed taking into account only those variables that are related to it, whether this relationship is linear or not. Using feature selection for this approach can be advantageous because it greatly reduces the number of preliminary imputations required which, in turn, greatly minimizes the need to contaminate the original data set with what are often overly simplified guesses (e.g., mean or mode, depending on the type of variable under consideration). Besides, depending on the form of the model chosen to generate predictions for the missing values, using feature selection can greatly assist in generating more robust missing values estimates because this pre-processing step can alleviate issues surrounding unreliable parameter estimates in the presence of multicollinearity and, consequently, abrupt changes in missing value estimates even with mild changes in the data used to train the prediction model. In general, the proposed scheme allows the optimization approach to move towards simpler, local convergence criteria than what is used in similar tree-based approaches and reduce the computational cost of the MVI scheme.

This work is an extension of Dávila and Rosado (2017), a previously published conference proceedings. In Dávila and Rosado (2017) a feature selection method was chosen arbitrarily and used in the proposed imputation scheme, whereas, in this paper a more exhaustive evaluation of various feature selection methods was performed in order to select the best option. Also, Rosado et al. did not mention the parameter tuning of the former TI-FS. Since then, numerous changes were made to the parameters of the method, new stopping rules were adopted, and more than twice the amount of data sets were considered in both the feature selection stage as well as in the evaluation of the imputation scheme.

The organization of this paper is as follows: Section 2 reviews some of the best MVI schemes and feature selection methods in the literature and describes how the TI-FS compares and contrasts to these methods. Sections 3 and 4 provide the conceptual framework and development of the proposed imputation scheme. Next, the performance of TI-FS was compared against K-nearest neighbors (Kowarik and Templ, 2016) and missForest (Stekhoven and Bülmann, 2012) in Section 5 using simulated and publicly available data sets. Lastly, the advantages and limitations of TI-FS are discussed and some conclusion remarks are given in Section 6.

## 2 Literature Review

### 2.1 Missing Value Imputation

A wide array of imputation methods have been proposed in literature to deal with the problem of missing data. They encompass anything from simple, like univariate mean/mode imputation to more complex multivariate schemes that look for relationships among covariates. Many studies have compared the performance of imputation methods but, unfortunately, regardless of the simplicity or complexity of an imputation method; its execution will always depend on the fitness between the data set, imputation method, and characteristics of the missing data (Sim et al., 2015).

One of the most popular and, by far, the simplest is *mean and mode substitution*. In this method, the missing values of a numerical variable are replaced by the mean of the observed cases, while missing categorical values are replaced with the covariate's mode (Silva et al., 2011). Mean/mode imputation is easy-to-use, but it is depicted as inferior since it distorts the covariance structure of the data, leading to biased estimates (Rogier et al., 2006).

Another commonly used method is *regression imputation*. Here, the missing values are predicted from a linear regression equation using the information from the complete cases (Enders, 2010). That is, the variable with missing values becomes the response and the remaining variables are used to predict this missing values. If the relationship between the variable being imputed and the remaining variables is linear; then, the method will work reasonably well. Otherwise, it will fail to understand the relationship among variables. Additionally, regression imputation also produces biased results, overestimating the correlations between covariates, and it only works on numerical data.

*Multiple imputation (MI)* has also been proven effective in MVI. A popular approach used to implementing MI is regression modeling, also known as multiple imputation by chain equations (MICE) (Burgette and Reiter, 2010). MICE imputes the missing values given a conditional model per covariate. The problem with MICE comes when specifying the conditional models for large amounts of covariates with missing values, even more so, when complex interactions exist between them. Identifying these models could be an uneasy task since it is hard to adjust a model that will fit the information of the missing data and simultaneously have convergence with the estimates (López, 2005). MI compares to the proposed method in that they both perform numerous imputations of the missing values in the data. Also, in that they are both conditional approaches, but the proposed method is only conditional on those variables that have a statistically significant relationship with the variable under consideration.

A more recent approach to MVI is *K-nearest neighbors (KNN)*. It is a non-parametric method that imputes missing data based on the outcome of the $K$ (a user-defined constant) observations closest to the missing value. Missing data are replaced with observed values from donors with similar characteristics (Stekhoven and Bülmann, 2012). Different distance measures are used to

determine the similarity between the missing values and the observed data. The most popular distance measure is the Euclidean distance, which is given by the root of squared differences between a pair of observations. Other distances commonly used are: Manhattan, Minkowski, Supremum Singh et al. (2013) and Gower Gower (1971).

KNN has proven to be effective when analyzing mixed-type data at different missing ratios (Yeşilova et al., 2010). It is also an attractive approach due to its simplicity and effectiveness in a variety of imputation problems (Liao et al., 2014). However, one of the drawbacks of the KNN method is that it only imputes a missing value based on its KNN, which makes it a conditional approach (López, 2005). Also, it is not clear which value of $K$ should be used. KNN and TI-FS are both conditional approaches. In KNN, donors represent observations with similar characteristics, whereas in TI-FS, a random forest is trained only on those covariates selected by the FS method to have a statistically significant difference with the covariate being imputed.

Tree-based MVI techniques are also widely used in mixed-type data sets with complex interactions between variables. Decision trees are non-parametric supervised methods produced by algorithms that identify various ways of splitting a data set into branch-like segments. Their goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data (Pantanowitz and Marwala, 2009). *Classification and regression trees (CART)* analysis is a technique that uses trees for predicting both continuous and categorical variables, where a tree is built by recursively partitioning the data set into non-overlapping regions (branches) and, then, the tree is used to predict the missing value for the covariate being treated as the dependent variable (Breiman et al., 1984).

A *Random forest (RF)* is an ensemble of decision trees that performs, both, classification and regression by drawing $M$ bootstrap samples from the original training data, using each of these $M$ samples to build $M$ trees within the ensemble. They can be easily adapted to the task of MVI (Breiman, 2001). In fact, RF can work around missing values without imputing them because they can decide what to do on a split based on the best surrogate for the variable under consideration. For the purpose of this work,RF' variable importance scores (VIS) are an added bonus. A VIS measures how much the prediction error increases when out-of-bag (OOB) data for that variable is permuted while all others are left unchanged (Liaw and Wiener, 2002). For a single decision tree, the measure of variable importance proposed by Breiman et al. (1984) is given:

$$VI(x_j, T) = \sum_{t \in T} \Delta I(x_j, t), \tag{1}$$

where $\Delta I(x_j, t) = I(t) - p_L I(t_L) - p_R I(t_R)$ is the decrease in impurity due to an actual or potential (surrogate) split on numerical predictor $x_i$ at a node $t$ of the optimally pruned tree, $T$ and $p_L(p_R)$ denotes the proportion of cases assigned to the left (right) child node of $t$. Note that Equation (1) refers to a single decision tree. For ensembles of $M$ trees, the VIS of a predictor is obtained by averaging over all its VIS across the ensemble. The process of averaging across all VIS in the ensembles has a stabilizing effect that leads to a more robust estimator of variable importance.

The rfImpute algorithm described in Pantanowitz and Marwala (2009) is a blend of KNN and RF approaches. For continuous covariates, the imputed value is the weighted average of the non-missing observations, using proximities as weights. For categorical predictors, the imputed value is the category with the largest average proximity.

The method in literature that most closely resembles the proposed approach is the iterative RF imputation scheme known as ***missForest***. In a similar approach to the proposed method,

missForest can be used with any type of data: Numerical, categorical, or even mixed-type data (Stekhoven and Bülmann, 2012). The main difference between this approach and the proposed method is how the RF is built on the observed data. TI-FS only have available statistically significant covariates to build the RF. Meanwhile, missForest have all predictor variables in the data set, regardless of the dimensionality of the problem and regardless of whether these predictors have any relationship with the predictor being imputed. As a result, the TI-FS can be used as an approximation to missForest in scenarios where the computational complexity of the imputation problem becomes an issue and the relationships among predictors are known or can be assessed a priori.

## 2.2 Feature Selection

The selection of relevant features or *feature selection* (FS) is a topic that has grown in popularity in recent years with the increase of complex, high-dimensional data. The presence of redundant or irrelevant features constitutes a problem since it can degrade the performance of learners in terms of speed and predictive accuracy.

Many algorithms have been developed to measure how useful a variable is in a data set. The objective of FS is to select a small subset of features from the original data that will provide the most significant information from a target (Kira and Rendell, 1992). Therefore, FS allows the analyst to better understand the data, reduces computational requirements, improves predictor performance, and facilitates to identify which features are relevant to a specific problem (Chandrashekar and Sahin, 2014).

***Filter selection methods*** are the most common approach for FS. They apply variable ranking techniques as the criterion for variable selection (Shardlow, 2008). *Correlation-based feature selection (CFS)* is a filter method that uses a correlation-based heuristic to evaluate a subset of features (Hall, 2000).

*ReliefF* is another commonly used filter method, extended from the original Relief (Kira and Rendell, 1992), that can deal with multi-class problems and incomplete data. It weights the features based how well their values distinguish between instances that are near to each other (Robnik-Sikonja and Kononenko, 2003).

While the greedy nature of filter approaches does not consider the interaction with the learner or even the interaction among the features (Saeys et al., 2007), correlation-based algorithms are significantly faster than other selection methods and have high prediction accuracy when analyzing high dimensional data (Lei and Liu, 2003; Doshi and Chaturvedi, 2014). ReliefF algorithms are also good in detecting conditional dependencies, thus, they are robust and noise-tolerant (Robnik-Sikonja and Kononenko, 2003).

The ***wrapper approach*** uses the prediction performance of a given induction algorithm/ learner to assess the usefulness of subsets of features in the data (Guyon and Elisseeff, 2003). Here, FS is “wrapped" around the learner and does an exhaustive search for variables in the data. The subset with highest performance is then chosen (Kohavi and John, 1997).

A popular wrapper approach is FS using *Genetic Algorithms (GA)*. This method mimics the properties of biological evolution (e.g crossover, inheritance, mutation, and selection) applying heuristic search methods to optimize the amount of variables in a data set (Pantanowitz and Marwala, 2009). Wrapper methods can be computationally intensive (Mohamad et al., 2004) and prone to overfitting, introducing bias and increasing the classification error. Nevertheless, GAs have demonstrated to be effective at handling both small and high-dimensional data.

***Embedded methods*** are somewhat similar to wrappers, but they perform FS as part of

the learning process. They are specific to a given algorithm that learns which features best contribute to the performance of a model. The most common examples of embedded methods are regularization methods (e.g. Lasso and Ridge regressions) (Brownlee, 2014) and decision tree-based methods. Embedded methods have the advantage that they include the interaction with the learner and are less computationally intensive than wrapper methods.

*Artificial Contrast with Ensembles (ACE)* is an embedded FS method that uses RF to select the best features in a data set. ACE creates a traditional statistical inference setting by building $N$ RF of $M$ trees and calculating $N$ VIS for $2J$ covariates in the training data set. That is, $J$ predictor covariates and $J$ additional artificial covariates. An artificial covariate $x_j^*$ is simply a random permutation of the observed values of predictor $x_j$. This process is repeated for each feature until the set of $J$ artificial covariates has been generated.

Since these artificial predictors are random permutations of the original, they share the same marginal distributions, but they are by no means related to the response. The idea is that their VIS must be low since they are not related to the response, and, hence, they can be used to create a threshold to better understand when the magnitude of a VIS is indeed large. From each of the $N$ RF, a VIS is recorded for each predictor as well as a large quantile ($q$), often $q_{0.8}$ or higher, for the VIS of artificial covariates. At the end of this iterative process, a paired t-test is used to determine whether each of the predictors has a VIS that is larger than the large quantile from the artificial VIS. All predictors that show a statistically significant improvement over the artificial variables are selected as important; the remaining predictors are discarded (Tuv et al., 2009). A Bonferroni approach is used to control the type I error.

The R package *VSURF* (Genuer et al., 2015), is also an embedded method that uses RF as a mean to select important features. It is based in a two-strategy approach: (1) preliminary elimination and ranking and (2) variable selection. In the first step, the objective is to find important variables highly related to the response using RF VIS. In the second step, a series of embedded RF are modeled starting with a RF build with only the most important variable and ending with a model having all the variables selected in the first step. The smallest model (and hence its corresponding variables), having a mean OOB error less than the minimum mean OOB error plus its standard deviation is selected.

## 3 Methodology

This section presents the detailed description of TI-FS, and the data used. The premise behind TI-FS is that a variable can be imputed taking into account only those variables that are related to it. When a missing value in a specific variable must be imputed, the imputation algorithm might not need to make a large amount of preliminary imputations in all other covariates with missing values or carry out a computationally-intensive optimization routine until convergence in the MVI is obtained. The method is currently implemented in two phases: FS followed by MVI.

### 3.1 Proposed Approach

Algorithm 1 summarizes the main steps of the proposed approach. Let $D_{mis}$ be a $n \times p$-dimensional data set having missing values. By default, the algorithm uses a forest of 125 trees ($T$) unless specified by the user. The maximum amount of imputations, $k$, per missing record is also set to 30, unless the user specifies otherwise. This value $k$ was set to 30 to ensure convergence of the imputation (Comulada, 2015). The categorical and numerical impute change

**Algorithm 1:** Proposed Imputation Scheme.

**Input :** A data set $D_{mis}$ having missing values
**Output :** A data set $D_{imp}$, with all missing values imputed
**Require:** $T \leftarrow 125$; /*Number of trees in the RF.*/;
$k \leftarrow 30$; /*Maximum amount of imputations.*/;
$t_c \leftarrow 4$; /*Categorical impute change.*/;
$t_n \leftarrow 0.025$; /*Numerical impute change.*/;
**foreach** $X_i$ **do**
  $m_i \leftarrow \texttt{CountMissVal}(X_i)$ /*$M$ is a vector of the frequency of missing values in each $X_i$.*/;
**end**
$D_0^{imp} \leftarrow \texttt{MeanMode}(D_{mis})$ /*Initial imputation using mean/mode.*/;
**foreach** $x_i$ **do**
  $f_i \leftarrow \texttt{GA}(D^i_{imp,0} \sim D^{-i}_{imp,0}$ /*Run feature selection.*/;
**end**
$F \leftarrow \texttt{CreateIncidenceMatrix}(F)$ /*Create Important variables incidence matrix.*/;
$O \leftarrow \texttt{Sort}(M)$ /*Vector of indices of columns in $D_{mis}$ sorted in increasing order of missing values.*/;
**foreach** $X_i \in D_{mis}$ **do**
  $cols \leftarrow \texttt{which}(f_i == 1) \cup \texttt{which}(\texttt{ColNames}(D_{mis}) ==$ "$Y$") /*Important variables and overall $Y$ of data.*/;
  $x_i^{obs} \leftarrow \texttt{NamesCompleteCases}(X_i)$ /*Row names of observed values in $X_i$.*/;
  $x_i^{mis} \leftarrow \texttt{NamesInCompleteCases}(X_i)$ /*Row names of missing values in $X_i$.*/;
  $X_i^{mis} \leftarrow X_i[x_{mis}]$ /*Missing values in $X_i$.*/;
  $D_{train} \leftarrow D_0^{imp}[x_i^{obs}, cols]$ /*Training sample.*/;
  $D_{test} \leftarrow D_0^{imp}[x_i^{mis}, cols]$ /*Testing sample.*/;
  **foreach** *record r in* $X_i$ **do**
    **while** $j$ *in* $1:k$ *or* $\delta_n > t_n$ *or* $\delta_c \neq t_c$ **do**
      $j = 1$;
      $rF \leftarrow \texttt{randomForest}(D_{train}[X_i] \sim D_{train}$, T) /*Fit RF.*/;
      $P \leftarrow \texttt{Predict}(X_i[x_i^{mis}])$;
      $\delta_n \leftarrow \texttt{ChangeInImpNum}(P_{new}, P_{old})$;
      $\delta_c \leftarrow \texttt{ChangeInImpCat}(P_{new}, P_{old})$ ;
      $j = j + 1$;
    **end**
  **end**
  **if** *is.numeric*$(X_i)$==*TRUE* **then**
    $X_i^{imp} = \texttt{mean}(P_{new}, P_{old})$;
  **end**
  **else**
    $X_i^{imp} = P_{new}$
  **end**
**end**
**Return** $D_{imp}$ /*Imputed data set.*/

thresholds, $t_c$ and $t_n$, are also required parameters within the algorithm. $t_c$ is set to 4, meaning that the imputation of record $r$ in a categorical missing variable $X_i$ stops after 4 unchanging consecutive imputations. In a similar way, the numerical impute change threshold ($t_n$) is set to 2.5%, meaning that the imputation for a record $r \in X_i$ stops when the percentage difference between the actual imputation ($P_{new}$) and an old one ($P_{old}$) is 2.5% or less.

Variables with missing values are first identified in the incomplete data set $D_{mis}$. A vector `M` is created with the amount of missing values in each missing variable $X_i$. Afterwards, an initial guess of the missing values in $D_{miss}$ is carried out using `mean/mode` imputation, prior to FS. The statistically significant variables for each $X_i \in D_{mis}$ are then determined using genetic algorithms (GA). An important variables incidence matrix `F` is created as a result. The columns in `F` refer to the variables with missing values, and the rows of the matrix refer to all variables in the data set. In this incidence matrix, a value of zero in element $i, j$ implies predictor $i$ was not detected as to have a significant relationship with predictor $j$. Otherwise, element $i, j$ would have a value of one, portraying a significant relationship between variables $i$ and $j$. The imputation of $X_i's$ is carried out in increasing order of missing values. Therefore, `O` is the vector of indices of columns in $D_{mis}$, sorted in increasing order of missing values. `O` indicates the order in which missing variables $X_i$ are imputed in the data set.

A RF of $T$ trees is built for each variable with missing values $X_i$, treating the vector of its observed values $D_{train}[X_i]$ as the response. The data set used to train the RF ($D_{train}$) includes the response of the overall supervised learning scenario, $Y$, as well as all other predictors that have been selected by the FS algorithm, as indicated by the elements that are equal to 1 in column $j$ of incidence matrix `F`. The RF draws $T$ bootstrap samples of $D_{train}$ to build the $T$ decision trees in the ensemble. The trained RF `rF` is then used to predict the missing values in $X_i$. The testing sample $D_{test}$, consists of $Y$ and the important variables of $X_i$ and it is used to predict $X_i$'s missing values.

This process is repeated $k$ times or until a stopping criterion $t$ is met. The change in imputation of numerical variables $\delta_n$, is the percentage of difference between the new imputation of a missing value, $P_{new}$ and its previous, as given by $P_{old}$. When $\delta_n$ is less than or equal to $t_n$, the algorithm stops imputing values for that record $r$ of $X_i$ and moves on to the next one. In the case of categorical variables, if the new imputed value has not changed in the last $t_c$ iterations, then, the algorithm stops the imputation process for that record and moves to the next $r$ in $X_i$. Finally, the overall imputation of numerical missing values, $X_i^{imp}$, is given by the average of the $j$ imputations used in $\delta_n$. Additionally, the final imputation for a categorical missing value is given by its last imputation.

### 3.2 Data

Different data sets were used throughout this work to evaluate the FSn, the parameter tunning of the RF imputation and the final comparison of the proposed imputation scheme with KNN and missForest. These data sets are divided into three groups mainly: Publicly available data, simulated scenarios and a special case study data set called Endometriosis Patient Registry (EPR).

Table 1 gives a general description of all data sets. *Cleveland Heart Disease* (Detrano et al., 1989), *Breast Cancer Wisconsin* (Wolberg and Mangasarian, 1990) and *Sylva Ecology* (US Forest Service, 2006) are publicly available at the UCI machine learning repository (Lichman, 2013). The simulated data sets used have: (1) linear relationships (Tuv et al., 2009)-*LinClass145, LinReg203*, (2) nonlinear relationships (Friedman, 2001)-*NonLinReg70, NonLinReg38, NonLin-*

Table 1: General description of data sets.

| Data set | Records | Num. Attr. | Cat. Attr. | Variables with MV's | Missing | Response |
|---|---|---|---|---|---|---|
| Heart disease | 303 | 5 | 8 | 2 | 2% | Class |
| Breast Cancer | 699 | 0 | 10 | 1 | 2% | Class |
| Sylva | 13,085 | 51 | 177 | 0 | 0% | Class |
| SimOriginal | 500 | 4 | 6 | 0 | 0% | Num |
| LinReg203 | 500 | 151 | 52 | 0 | 0% | Num |
| LinClass145 | 500 | 86 | 59 | 0 | 0% | Class |
| NonLinReg70 | 500 | 70 | 0 | 0 | 0% | Num |
| NonLinReg38 | 500 | 27 | 11 | 0 | 0% | Num |
| NonLinReg125 | 500 | 77 | 48 | 0 | 0% | Num |
| EPR | 2,763 | 5 | 94 | 25 | 14% | Class |

Table 2: SimOriginal data structure. Simulated data includes 500 observations in 10 mixed-type data predictors and one numerical response variable.

| $X_{00} \sim \text{Unif}(-0.25, 0.25)$ | $X_{01} \sim N(0,1)$ |
|---|---|
| $X_1 = X_{00} + X_{01}$ | $X_7 \sim \text{Unif}(1,3)$ |
| $X_2 = 2 * X_1 + X_{01}$ | $X_8 \sim \text{Unif}(1,5)$ |
| $X_3 = X_1 + X_2 + X_{01}$ | $X_9 \sim \text{Unif}(1,7)$ |
| $X_4 = X_1 * X_2 + X_{01}$ | $X_{10} \sim \text{Unif}(1,10)$ |
| $X_5 = \begin{cases} \text{Unif}(2,4) & X_1 \geq 0 \\ 1 & \text{elsewhere} \end{cases}$ | $Y \sim 3X_1 + X_2 + X_{00}$ |
| $X_6 \sim \text{Unif}(1,2)$ | |

*Reg125* and (3) *SimOriginal* which was simulated using R software (Team, 2016).

The structure of the *SimOriginal* data set is shown in Table 2. It was simulated so that half of the predictors are related ($X_1 - X_5$), while the remaining half are independent ($X_6 - X_{10}$). The value of predictor $X_2$ depends on the value of predictor $X_1$ plus random noise based on the standard normal distribution, $N(0,1)$. Predictors $X_3 - X_5$ also depend on the value of $X_1$ and so on. In addition, predictors $X_3$ and $X_4$ further depend on $X_2$. For $X_3$, the relationship with $X_2$ is additive, whereas its relationship with $X_4$ is multiplicative. Finally, the response was generated based on an additive model using $X_1$, $X_2$ and a random uniform noise.

The Endometriosis Patient Registry (EPR) is a data set from the Endometriosis Research Program (ERP) at the Ponce School of Medicine and Health Sciences, is relatively large with a moderate number of missing values. This registry gathers information of women with endometriosis-related symptoms, some of which chose to be diagnosed via an invasive surgical procedure (e.g. laparoscopy, laparotomy). It includes data on demographical information, endometriosis-related symptoms, pre-existing conditions, lifestyle choices, and family and medical history for a total of 99 different variables and 2,763 records.

The EPR's main challenge is the fact that it has more than 37,000 (14%) missing values. If any record with missing values were to be discarded, there would be zero records left. This issue comes up from the fact that this data was collected using a survey that was subject to changes over a ten-year period (e.g. some questions were added, some questions were removed). Also, it is the most complete data repository for endometriosis patients in Puerto Rico. It took a significant amount of time and effort to gather and, thus, it is key for advancing the knowledge of endometriosis in Puerto Rican patients.

## 4 Analysis

TI-FS consists of two phases: (1) selecting the important features of each missing variable in the data set and (2) imputing the missing variables based on the significant variables chosen by the FS. Figure 1 depicts an overview of the evaluation done throughout this work. In order to develop TI-FS, five different FS methods were considered and evaluated in a cross-validation (CV) setting and an extensive parameter tuning was carried out to determine the most suitable combination of parameters for the RF imputation. Lastly, its performance was compared to KNN and missForest.

### 4.1 Evaluation of FS Methods

Five FS methods (ACE, CFS, ReliefF, GA and VSURF) were evaluated and the most suitable was used in TI-FS. The performance of the FS methods was assessed using five-fold CV on a RF model. In addition, a second evaluation was carried out using the simulated data sets, since their structure was known. Seven data sets (discussed in Section 3.2 of different sizes were used in this evaluation. Thirty bootstrap samples of each of the seven data sets were created, therefore, the five-fold CV was carried out 30 times for each FS method and each data set. All of the experiments in this phase were performed using the R statistical software (Team, 2016).

The *ACE* method used in this evaluation is a modified version of the original ACE by Tuv et al. (2009). This Tuv et al. (2009) version uses gradient boosted trees (GBT) to obtain the variable scores, whereas the original ACE uses RF. The *CFS and ReliefF* FS methods were

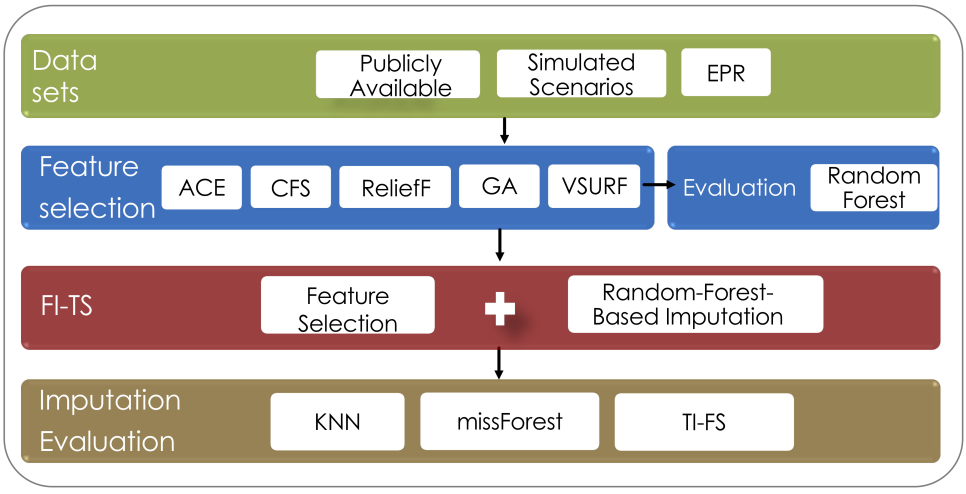


Figure 1: Flow chart of proposed MVI approach.

implemented using the R package `FSelector` (Romanski and Kotthoff, 2018). The *VSURF* method was implemented using the R package `VSURF` (Genuer et al., 2015) as well. Finally, the *GA* FS algorithm was executed using the R package `caret` (Kuhn, 2020).

**Performance Measures** Four performance measures were evaluated in a five-fold cross-validation setting: (1) *accuracy*, (2) *best subset size*, (3) *run time* and (4) *overall desirability*. In classification models, the *accuracy* is the fraction of instances that are correctly predicted, however, the accuracy of a regression model is given by the prediction error (PRESS). PRESS values were scaled to a range between 0 and 1 and its complement was calculated to convert them into accuracy values. Consequently, values close to one are preferred for this scaled measure.

*Run time* denotes the CPU time, in seconds, taken to run the algorithm. A faster FS method is desired; hence, lower run time values are preferred. The *Best subset size* depicts the number of important variables selected by the FS method. Smaller subsets will lead to less complex models and, thus, smaller subsets are preferred.

Since various performance measures may favor different methods, a *desirability function* was used to determine the top performer. The overall desirability function ($D_{FS,1}$) combines the previous measures and gives each one of them different weights based on their relative importance: Accuracy (60%) > run time (30%) > best subset (10%) as in Equation (2):

$$D_{FS,1} = 0.6 \times (Accuracy) + 0.3 \times (1 - Runtime') + 0.1 \times (1 - BestSubset'). \tag{2}$$

Note that all the performance measures were scaled between 0 and 1 before calculating the desirability function using Equation (3):

$$P_i' = \frac{P_i - \min(P_i)}{\max(P_i) - \min(P_i)}. \tag{3}$$

In addition, the *Sensitivity*, *Specificity*, *Accuracy* and overall *Desirability function* were calculated for the simulated data sets since their important variables were known. The sensitivity was calculated as the proportion of correctly identified important variables and the specificity as the proportion of irrelevant variables. The overall desirability ($D_{FS_2}$) combines the previous measures in Equation (4) and gives each one of them different weights based on their relative importance: Sensitivity (50%) > specificity (25%) > accuracy (25%). Equation (4) was scaled to values between 0 and 1 as well:

$$D_{FS,2} = 0.5 \times Sensitivity + 0.25 \times Specificity + 0.25 \times Accuracy. \tag{4}$$

## 4.2 Evaluation of RF Imputation

Additional experiments were carried out in order to improve the RF-based imputation performance. Specifically, various factors were taken into account for the evaluation of the stopping criteria in the imputation of the missing values in $X_i$. This algorithm was implemented using R statistical software as well as the RF function available in the R package `randomForest` (Liaw and Wiener, 2002).

Figure 2 portrays the multiple factors evaluated for the imputation stopping criteria and the overall imputation. Five different factors were considered in this evaluation: (1) missing ratio, (2) impute change, (3) stage, (4) stop rule, and (5) imputation strategy. Since the efficiency of an imputation technique depends on the amount of missing values present in a data set, the experiments were performed using randomly generated missing ratios of 5%, 10%, 15% and 20%.

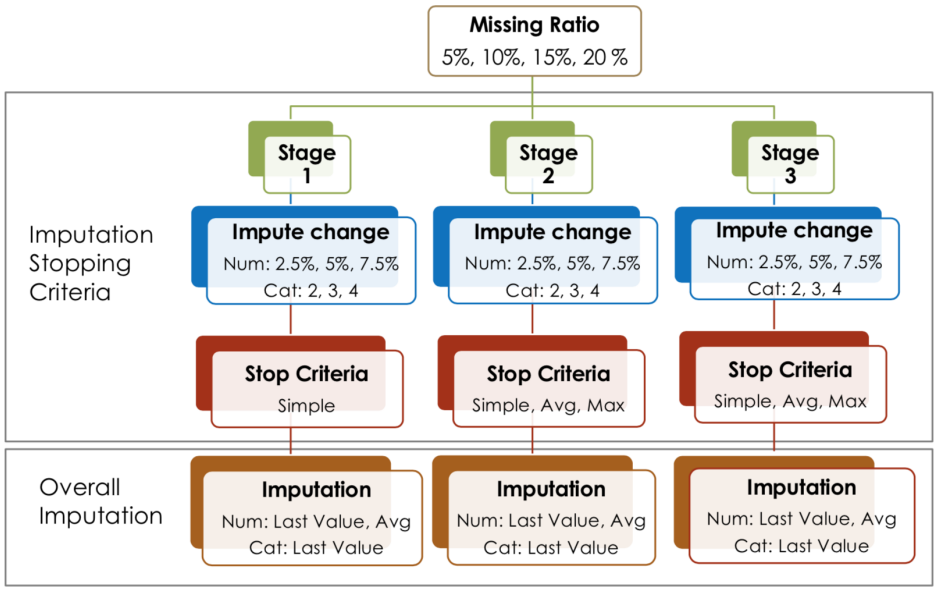


Figure 2: RF imputation parameter tuning. Diagram of the factors and their corresponding levels evaluated for the imputation stopping criteria and the overall imputation.

The stopping criteria is calculated for numerical variables, $\delta_n$, as the percentage of difference between the new imputation of a missing value and an old imputation (Equation (5)). This difference is compared against a threshold called *impute change* ($t_n$). The algorithm will keep imputing the missing value as long as $\delta_n > t_n$. The idea here is that $t_n$ should be a small value in order to say that the imputed value is no longer changing. Thus, the following values for the *impute change* threshold were evaluated: 2.5%, 5% and 7.5%:

$$\delta_n = \frac{x_{new} - x_{old}}{x_{old}}. \tag{5}$$

In the case of categorical variables, if the new imputed value stays the same in the last $\delta_c$ iterations, then, the algorithm stops imputing for that record and moves on to the next record $r$ in $X_i$. Three options were also evaluated for $t_c$, where the imputation value did not: Change in the last two iterations ($t_c$=2), change in the last three iterations ($t_c$=3) and did not change in the last three iterations ($t_c$=4).

Two factors were considered to decide how $\delta_n$ is calculated in the algorithm: (1) *stage* and (2) *stop rule*. Three stages were evaluated: Stage 1, 2, and 3, denoting the amount of imputation differences considered for the final impute change in the iteration. These stages go along with three stopping criterias: Simple, average and maximum, which depict the aggregation of these stages to obtain the final imputation change. Table 3 shows some of the combinations of *stages* and *stop rules* along with the sample equations used to calculate $\delta_n$ in the experiments. For example, if the combination stage 2/average is used, then, $\delta_n$ is calculated as the average of the difference between imputations $j$ and $j-1$ and imputations $j-1$ and $j-2$. The same rationale was used for the rest of the combinations.

Finally, two options were evaluated for the overall imputation value of record $r$ in numerical missing variable $X_i$: Last value, meaning that the final imputed value is $x_j^{imp}$, and average, which implies that the final imputed value is the average of the imputed values considered in the

Table 3: Examples of equations used to calculate $\delta_n$ in the parameter tuning.

| Stage | Stop Criteria | $\delta_n$ |
|---|---|---|
| 1 | Simple | $\frac{(x_j^{imp} - x_{j-1}^{imp})}{x_{j-1}^{imp}}$ |
| 2 | Average | $\frac{\frac{(x_j^{imp} - x_{j-1}^{imp})}{x_{j-1}^{imp}} + \frac{(x_{j-1}^{imp} - x_{j-2}^{imp})}{x_{j-2}^{imp}}}{2}$ |
| 3 | Maximum | $\max\left\{\frac{(x_j^{imp} - x_{j-1}^{imp})}{x_{j-1}^{imp}}, \frac{(x_{j-1}^{imp} - x_{j-2}^{imp})}{x_{j-2}^{imp}}, \frac{(x_{j-2}^{imp} - x_{j-3}^{imp})}{x_{j-3}^{imp}}\right\}$ |

calculation of $\delta_n$. If $X_i$ is a categorical variable, then, the final imputed value ($x_j^{imp}$) is the last value imputed. Overall, forty two different combinations were evaluated for each missing ratio, for a total of 168 combinations.

**Performance Measures** Both, regression and classification performance measures were evaluated due to the mixed-typed nature of the data sets. Four performance measures for numerical variables were evaluated: (1) *coefficient of determination* ($R^2$), (2) *normalized root mean squared error* (NRMSE), (3) *index of agreement* ($d_2$) and (4) *overall numerical desirability* ($D_n$). $R^2$, *NRMSE*, and $d_2$ were implemented using the `hydroGOF` R package (Zambrano-Bigiarini, 2020).

The *overall numerical desirability* ($D_n$) is an additive function that combines all performance measures and treats them as equally important (Equation (6)):

$$D_n = R^2 + d_2 + (1 - \text{NRMSE}). \tag{6}$$

Note that the complement of NRMSE is used in order to reflect that larger values of the desirability function are preferred.

The *classification error* (E), *area under precision-recall curve* (AUPRC) *kappa statistic* ($\kappa$) and *overall categorical desirability* ($D_c$) were used to evaluate the performance of categorical variables. The *classification error* denotes the proportion of sample cases incorrectly classified. *AUPRC* and the *kappa statistic* were calculated using the `PRROC` (Grau and Keilwagen, 2018) and the `psych` (Revelle, 2019) R packages, respectively.

In the same way as with the numerical variables, the *overall categorical desirability* ($D_c$) was assessed considering each term as equally important (Equation (7)):

$$D_c = \kappa + \text{AUPRC} + (1 - E). \tag{7}$$

Lastly, in order to evaluate both, numerical and categorical variables at the same time, an overall desirability function ($D_o$) was calculated as the summation of the individual numerical and categorical desirabilities. Both are equally weighted as in:

$$D_o = D_n + D_c. \tag{8}$$

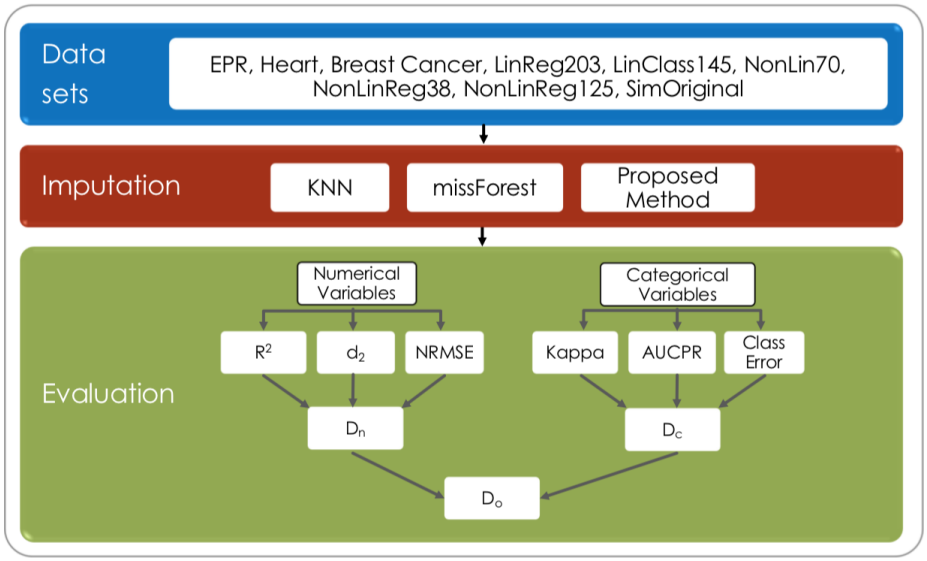


Figure 3: Flow chart of the performance evaluation of the MVI schemes.

### 4.3 Evaluation of Imputation Methods

The performance of TI-FS was compared to KNN and missForest. Figure 3 displays a general overview of the MVI evaluation. Nine data sets (BreastCancer, Heart, EPR, LinReg203, LinClass145, NonLinReg70, NonLinReg125, NonLinReg38 and SimOriginal), described in Section 3.2, were used in this final evaluation. Thirty bootstrap samples were generated for each data set and missing values were randomly created for each one of them. These missing values were simulated at 5%, 10%, 15% and 20% missing ratios.

KNN and missForest were implemented using the R packages `VIM` (Kowarik and Templ, 2016) and `missForest` (Stekhoven, 2013), respectively. The performance measures used for the evaluation of the imputation methods are the same as the ones explained in Subsection 4.1.

## 5 Results

All experiments were run and the performance of TI-FS was assessed. This section summarizes the major results of the for the evaluation of the FS method and the final assessment of the performance of TI-FS against KNN and missForest.

### 5.1 TI-FS

**FS** Table 4 shows the aggregated results for the five-fold CV used in the selection of the FS method. In Table 4, columns 3-6 show the average value of the performance metrics on each FS method across the seven data sets. Bold values in the $D'_{FS,1}$ column represent the largest normalized desirability. Figure 4 also shows the average desirability of the FS methods, including standard error bars for 30 replicates. Overall, CFS performed the best in 4 out of 7 data sets (57% of the evaluated cases).

Table 4: FS five-fold CV results. $D'_{FS,1}$ values were averaged across replicates and normalized with respect to the largest desirability of each data set.

| Data set | FS Method | Accuracy | Run Time | Best Subset | $D'_{FS,1}$ |
|---|---|---|---|---|---|
| BreastCancer | CFS | 0.9834 | 0.0013 | 8 | 0.8157 |
| | GA | 0.9824 | 2.1136 | 7 | 0.7233 |
| | ACE | 0.9853 | 0.0883 | 10 | 0.7548 |
| | ReliefF | 0.9675 | 0.0603 | 6 | **1.0000** |
| | VSURF | 0.9839 | 1.7940 | 6 | 0.7613 |
| EPR | CFS | 0.7819 | 0.2680 | 11 | 0.9159 |
| | GA | 0.8934 | 7.0413 | 70 | 0.8305 |
| | ACE | 0.7945 | 0.3943 | 28 | 0.7047 |
| | ReliefF | 0.6782 | 0.5435 | 10 | 0.3755 |
| | VSURF | 0.7278 | 59.5774 | 12 | **1.0000** |
| Heart | CFS | 0.7497 | 0.0220 | 5 | 0.5775 |
| | GA | 0.8077 | 0.7780 | 10 | 0.6229 |
| | ACE | 0.8112 | 0.0057 | 8 | 0.6224 |
| | ReliefF | 0.9735 | 0.0343 | 7 | **1.0000** |
| | VSURF | 0.8016 | 0.1663 | 7 | 0.6430 |
| LinReg203 | CFS | 0.4579 | 0.0350 | 2 | **1.0000** |
| | GA | 0.7064 | 1.0043 | 86 | 0.9286 |
| | ACE | 0.7465 | 0.0023 | 73 | 0.8985 |
| | ReliefF | 0.4420 | 0.0613 | 78 | 0.9288 |
| | VSURF | 0.7673 | 7.0063 | 6 | 0.9910 |
| LinClass145 | CFS | 0.7196 | 0.0300 | 9 | **1.0000** |
| | GA | 0.3981 | 0.9180 | 85 | 0.6589 |
| | ACE | 0.7110 | 0.0463 | 29 | 0.8131 |
| | ReliefF | 0.6658 | 0.0427 | 52 | 0.7718 |
| | VSURF | 0.7045 | 5.8063 | 7 | 0.7942 |
| NonLinReg70 | CFS | 0.6588 | 0.0253 | 5 | **1.0000** |
| | GA | 0.8022 | 0.8507 | 38 | 0.7779 |
| | ACE | 0.8256 | 0.0243 | 22 | 0.8011 |
| | ReliefF | 0.6915 | 0.0683 | 17 | 0.8084 |
| | VSURF | 0.8396 | 3.2400 | 7 | 0.9076 |
| Sylva | CFS | 0.9852 | 0.9700 | 7 | **1.0000** |
| | GA | 0.9961 | 31.6630 | 147 | 0.7014 |
| | ACE | 0.9967 | 3.4700 | 59 | 0.7406 |
| | ReliefF | 0.9597 | 1.1637 | 18 | 0.6246 |
| | VSURF | 0.9949 | 334.8100 | 8 | 0.8399 |

Table 5: FS performance on simulated scenarios. $D'_{FS,2}$ values were averaged across replicates and normalized with respect to the largest desirability of each data set.

| Data set | FS Method | Sensitivity | Specificity | Accuracy | $D'_{FS,2}$ |
|---|---|---|---|---|---|
| LinClass145 | ACE | 0.4467 | 0.8321 | 0.7110 | 0.7737 |
| | CFS | 0.4644 | 0.9849 | 0.7196 | **1.0000** |
| | GA | 0.6867 | 0.4274 | 0.4543 | 0.8159 |
| | ReliefF | 0.4956 | 0.6582 | 0.6658 | 0.8217 |
| | VSURF | 0.1978 | 0.9718 | 0.7045 | 0.5889 |
| LinReg203 | ACE | 0.1667 | 0.6365 | 0.6273 | 0.8571 |
| | CFS | 0.0167 | 0.9903 | 0.9711 | 0.1000 |
| | GA | 0.1667 | 0.5008 | 0.4943 | **1.0000** |
| | ReliefF | 0.6250 | 0.6198 | 0.6199 | 0.7685 |
| | VSURF | 0.7500 | 0.9864 | 0.9818 | 0.7381 |
| NonLinReg70 | ACE | 0.6900 | 0.7472 | 0.7390 | 0.8326 |
| | CFS | 0.1967 | 0.9444 | 0.8376 | 0.6591 |
| | GA | 0.8633 | 0.5044 | 0.5557 | **1.0000** |
| | ReliefF | 0.3333 | 0.7756 | 0.7124 | 0.6812 |
| | VSURF | 0.4433 | 0.9611 | 0.8871 | 0.8410 |

Table 6: Normalized sum of desirability scores of the FS methods across all data sets.

| FS Method | $D_{FS,1}$ score | $D_{FS,2}$ score |
|---|---|---|
| ACE | 0.8456 | 0.8748 |
| CFS | **1.000** | 0.6247 |
| GA | 0.8311 | **1.0000** |
| ReliefF | 0.8732 | 0.8066 |
| VSURF | 0.9410 | 0.7699 |

As mentioned in Subsection 4.1, the simulated data sets (LinReg203, LinClass145 and NonLinReg70) were evaluated in more detail since their structure and important variables were known. Table 5 shows the results of this analysis, again, bold values in the $D'_{FS,2}$ column show the largest desirability. Figure 5 also displays the average desirability results including standard error bars for 30 replicates. GA performed better in 2 of 3 data sets (66% of the evaluated scenarios).

Table 6 portrays the normalized desirability score for each FS method. This score is given by the normalized sum of the desirability function across the data sets. The third column in Table 6 is the score across all data sets using the results from Table 4 ($D'_{FS,1}$), while column 4 shows the scores across the simulated data sets (see Table 5-$D'_{FS,2}$). These scores confirm the previous results, CFS and GA are the top FS methods. Note that the issue here is failing to

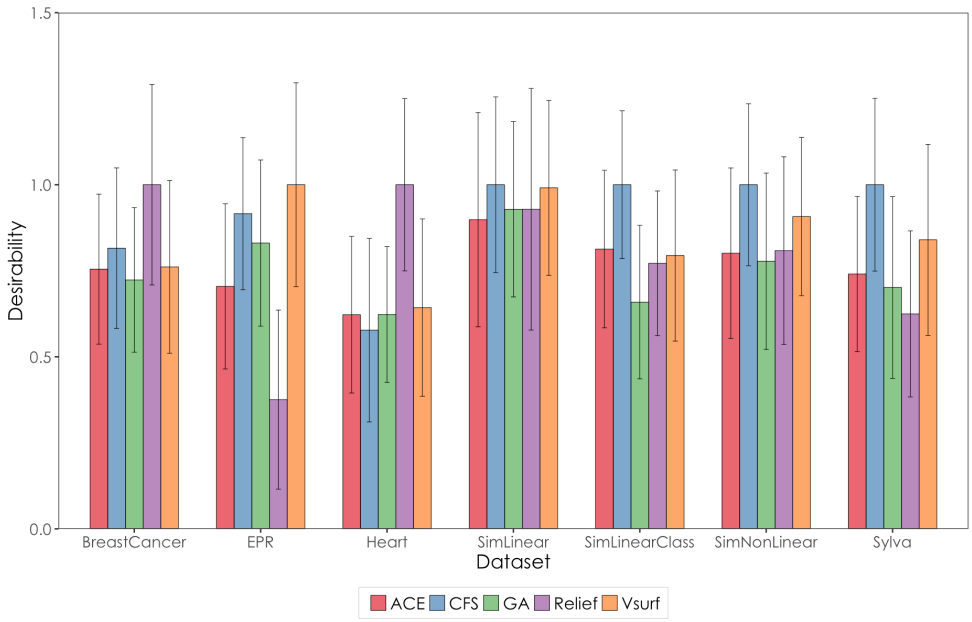


Figure 4: Average desirability and standard errors based on 30 replicates of FS methods.

detect important variables since TI-FS imputes missing variables strictly based on the important variables selected by the FS method. Selecting variables in excess might not have much of an effect on the overall execution of the imputation scheme, as trees can choose not to incorporate them within the prediction model.

**RF Imputation** The parameter tuning of the RF imputation (discussed in Subsection 4.2) was carried out using the top two FS methods, CFS and GA. Both methods were used in order to evaluate which FS method in fact allows for the best imputation performance. Table 7 shows the top 3 combinations of parameters for each missing ratio and FS method, including standard error bars of 30 replicates. Bold values in the column $D'_o$ represent the largest desirability. Figure 6 also portrays the top 3 performing combinations of parameters (using both GA and CFS) at each missing ratio in terms of the overall normalized desirability ($D'_o$). The majority of the $D'_o$ results in Figure 6 fall in the lower boxes of the plot, specifically in the lower left. This corner includes, for the most part, restrictive assumptions with regards to the impute difference calculation of numerical variables (stage 3) and its threshold ($t_n = 2.5\%$). These results suggest that restrictive combinations of parameters perform better with higher ratios of missing values.

In summary, the results in Table 7 confirm GA as the best choice for FS, given that it leads to the largest values of desirability ($D'_o$) in three out of four missing ratios (or 75% of the cases). Furthermore, the selected combinations of parameters for TI-FS are given in Table 8. This combination was chosen based on the premise that the proposed method is mostly focused on data with medium to high proportions of missing values. Thus, the proposed imputation scheme used GA as FS method along with the random-forest-based imputation having $t_n = 2.5\%$, $t_c = 4$, $\delta_n = \frac{x_j^{imp} - x_{j-3}^{imp}}{x_{j-3}^{imp}}$, with $x_n^{imp} = \frac{x_{j-3}^{imp} + x_j^{imp}}{2}$ being the final imputation of numerical missing values and $x_c^{imp} = x_j^{imp}$ the final imputation of categorical missing values.

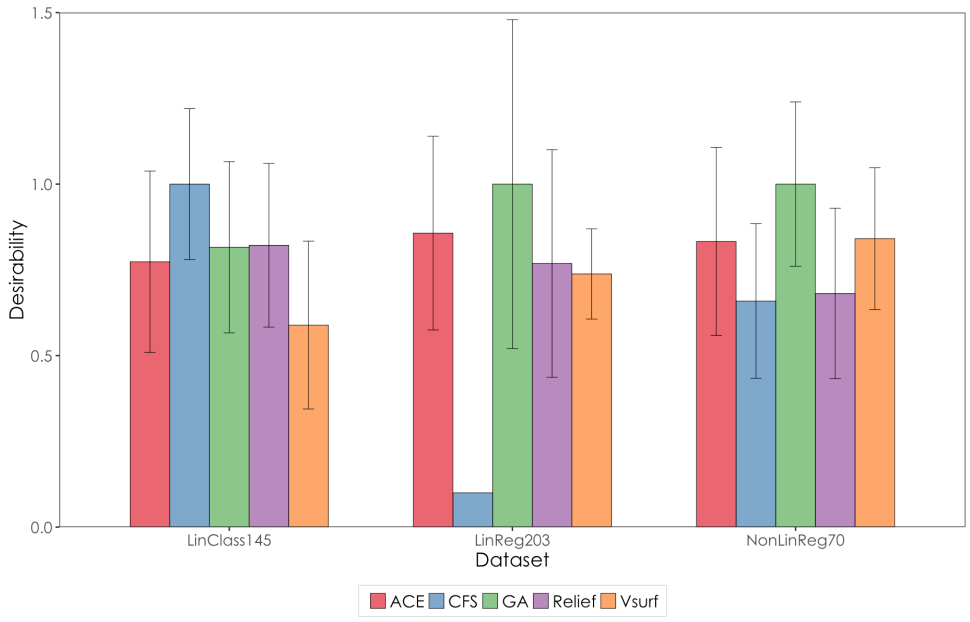


Figure 5: Average desirability and standard errors for 30 replicates of the FS methods on LinClass145, LinReg203 and NonLinReg70.

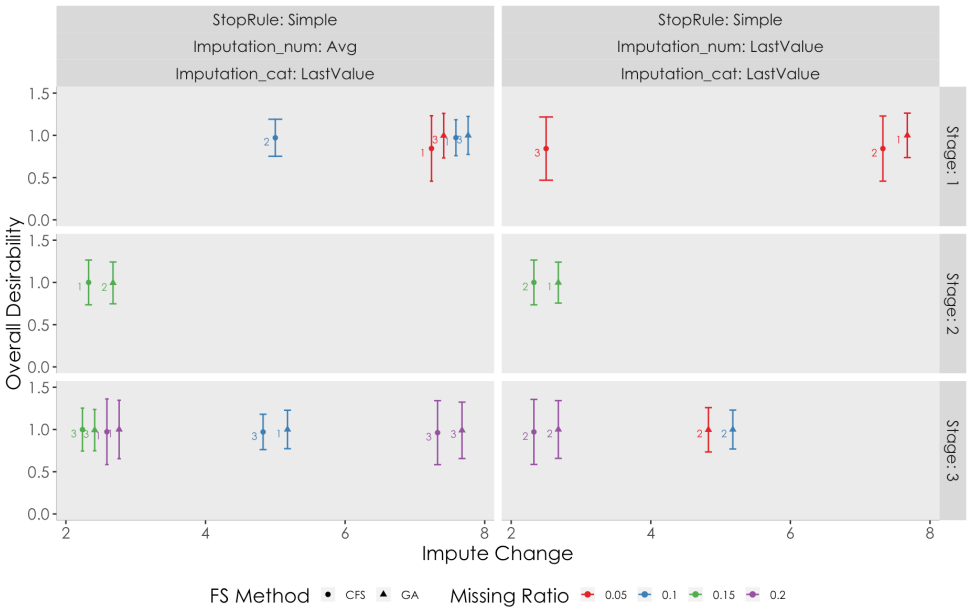


Figure 6: Top three performing combinations of CFS and GA per missing ratio.

Table 7: Best performing combinations of parameters for GA and CFS per missing ratio. $D'_o$ values were averaged across replicates and normalized with respect to the largest desirability for each combination of FS method and missing ratio.

| Missing Ratio (%) | FS Method | Impute Change (cat/num) | Stage | Stop Rule | Imputation (cat) | Imputation (num) | $D'_o$ |
|---|---|---|---|---|---|---|---|
| 5 | GA | 2/7.5 | 1 | Simple | LastValue | LastValue | **1.0000** |
| | | 3/5 | 3 | Simple | LastValue | LastValue | 0.9964 |
| | | 2/7.5 | 1 | Simple | LastValue | Avg | 0.9963 |
| | CFS | 2/7.5 | 1 | Simple | LastValue | Avg | 0.8447 |
| | | 2/7.5 | 1 | Simple | LastValue | LastValue | 0.8434 |
| | | 4/2.5 | 1 | Simple | LastValue | LastValue | 0.8432 |
| 10 | GA | 3/5 | 3 | Simple | LastValue | Avg | **1.0000** |
| | | 3/5 | 3 | Simple | LastValue | LastValue | 0.9993 |
| | | 2/7.5 | 1 | Simple | LastValue | Avg | 0.9992 |
| | CFS | 2/7.5 | 1 | Simple | LastValue | Avg | 0.9722 |
| | | 3/5 | 1 | Simple | LastValue | Avg | 0.9716 |
| | | 3/5 | 3 | Simple | LastValue | Avg | 0.9710 |
| 15 | CFS | 4/2.5 | 2 | Simple | LastValue | Avg | **1.0000** |
| | | 4/2.5 | 2 | Simple | LastValue | LastValue | 0.9997 |
| | | 4/2.5 | 3 | Simple | LastValue | Avg | 0.9988 |
| | GA | 4/2.5 | 2 | Simple | LastValue | LastValue | 0.9977 |
| | | 4/2.5 | 2 | Simple | LastValue | Avg | 0.9943 |
| | | 4/2.5 | 3 | Simple | LastValue | Avg | 0.9926 |
| 20 | GA | 4/2.5 | 3 | Simple | LastValue | Avg | **1.0000** |
| | | 4/2.5 | 3 | Simple | LastValue | LastValue | 0.9999 |
| | | 2/7.5 | 3 | Simple | LastValue | Avg | 0.9899 |
| | CFS | 4/2.5 | 3 | Simple | LastValue | Avg | 0.9728 |
| | | 4/2.5 | 3 | Simple | LastValue | LastValue | 0.9710 |
| | | 2/7.5 | 3 | Simple | LastValue | Avg | 0.9624 |

Table 8: Selected combination of parameters for TI-FS.

| Parameter | Setting |
|---|---|
| Impute change num | 2.5 |
| Impute change cat | 4 |
| Stage | 3 |
| Stop Rule | Simple |
| Imputation num | Avg |
| Imputation cat | Last Value |

Table 9: Normalized overall desirability ($D'_o$) of imputation methods. $D'_o$ values were averaged across replicates and normalized with respect to the largest desirability for each combination of data set and missing ratio.

| Data set | Method | Missing Ratio | | | |
|---|---|---|---|---|---|
| | | 0.05 | 0.10 | 0.15 | 0.20 |
| BreastCancer | KNN | 0.8109 | 0.7870 | 0.7888 | 0.7911 |
| | FS-TI | 0.4199 | 0.4203 | 0.4325 | 0.4699 |
| | missForest | **1.0000** | **1.0000** | **1.0000** | **1.0000** |
| EPR | KNN | 0.9387 | 0.9321 | 0.9167 | 0.9160 |
| | FS-TI | 0.6549 | 0.6181 | 0.6199 | 0.5968 |
| | missForest | **1.0000** | **1.0000** | **1.0000** | **1.0000** |
| Heart | KNN | 0.9621 | 0.9610 | 0.9628 | 0.9670 |
| | FS-TI | 0.7967 | 0.7950 | 0.7612 | 0.8638 |
| | missForest | **1.0000** | **1.0000** | **1.0000** | **1.0000** |
| LinClass145 | KNN | 0.7719 | 0.7296 | 0.7234 | 0.7071 |
| | FS-TI | **1.0000** | **1.0000** | 0.9438 | 0.8949 |
| | missForest | 0.9996 | 0.9763 | **1.0000** | **1.0000** |
| LinReg203 | KNN | 0.9270 | 0.9342 | 0.9352 | 0.9293 |
| | FS-TI | 0.9480 | 0.9485 | 0.9889 | 0.9739 |
| | missForest | **1.0000** | **1.0000** | **1.0000** | **1.0000** |
| NonLinReg125 | KNN | 0.7965 | 0.7722 | 0.7261 | 0.7128 |
| | FS-TI | **1.0000** | 0.9684 | **1.0000** | 0.9480 |
| | missForest | 0.9911 | **1.0000** | 0.9791 | **1.0000** |
| NonLinReg38 | KNN | 0.8515 | 0.8139 | 0.7809 | 0.7609 |
| | FS-TI | 0.9704 | 0.9498 | 0.9614 | 0.9474 |
| | missForest | **1.0000** | **1.0000** | **1.0000** | **1.0000** |
| NonLinReg70 | KNN | 0.7140 | 0.6653 | 0.6336 | 0.6145 |
| | FS-TI | **1.0000** | 0.9769 | 0.9618 | 0.9420 |
| | missForest | 0.9944 | **1.0000** | **1.0000** | **1.0000** |
| SimOriginal | KNN | 0.2926 | 0.9465 | 0.3191 | 0.9152 |
| | FS-TI | **1.0000** | **1.0000** | **1.0000** | **1.0000** |
| | missForest | 0.4203 | 0.9532 | 0.4274 | 0.9780 |

## 5.2 Performance of Imputation Methods

Table 9 shows the average normalized overall desirability of the three imputation methods. Bold values represent the largest desirability of each data set and missing ratio. Figure 7 also compares the average $D'_o$ results including standard error bars based on 30 replicates.

TI-FS outperforms KNN in the simulated scenarios, which have complex linear, non-linear, additive and multiplicative relationships between variables. Figures 8–9 suggest that TI-FS performs better on imputing categorical variables than on numerical variables, specially at higher missing ratios (15% and 20%), where there is less than a 4% difference between TI-FS and missForest. It must noted that, in these scenarios, TI-FS uses between 42% and 59% of the variables that are available to missForest for imputation.

In general, missForest was the top performer. However, TI-FS still a reasonable approxima-

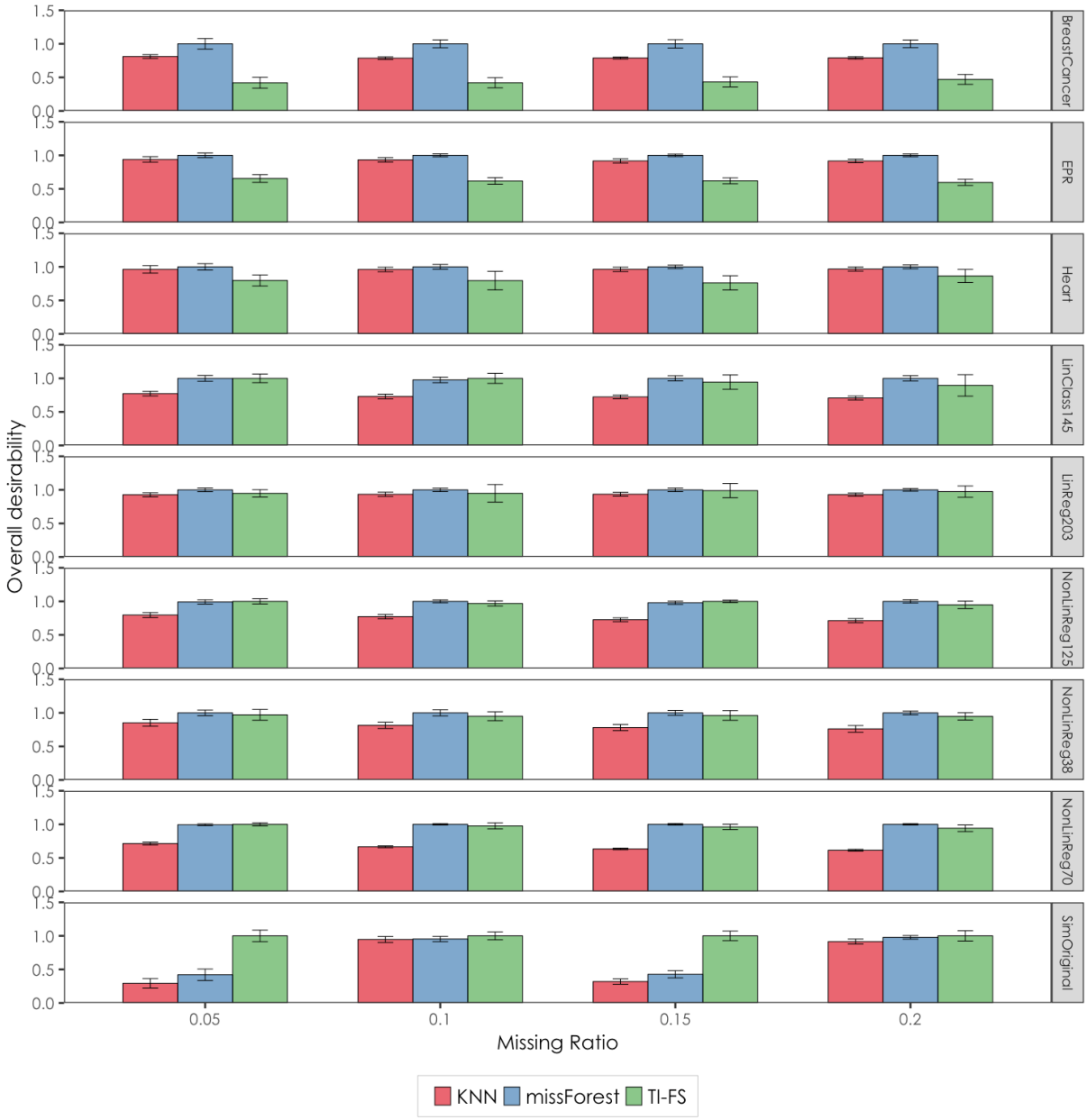


Figure 7: Average performance of the imputation methods based on the normalized overall desirability.

tion if considered that it had, on average, 60% of the total predictors available to carry out the imputations, whereas the RF in missForest treated all variables as candidate splitters. Table 10 shows that the performance of TI-FS is comparable to missForest in the simulated scenarios, with no more than 3.6% difference, yet, it had approximately half of the variables available to execute the imputations.

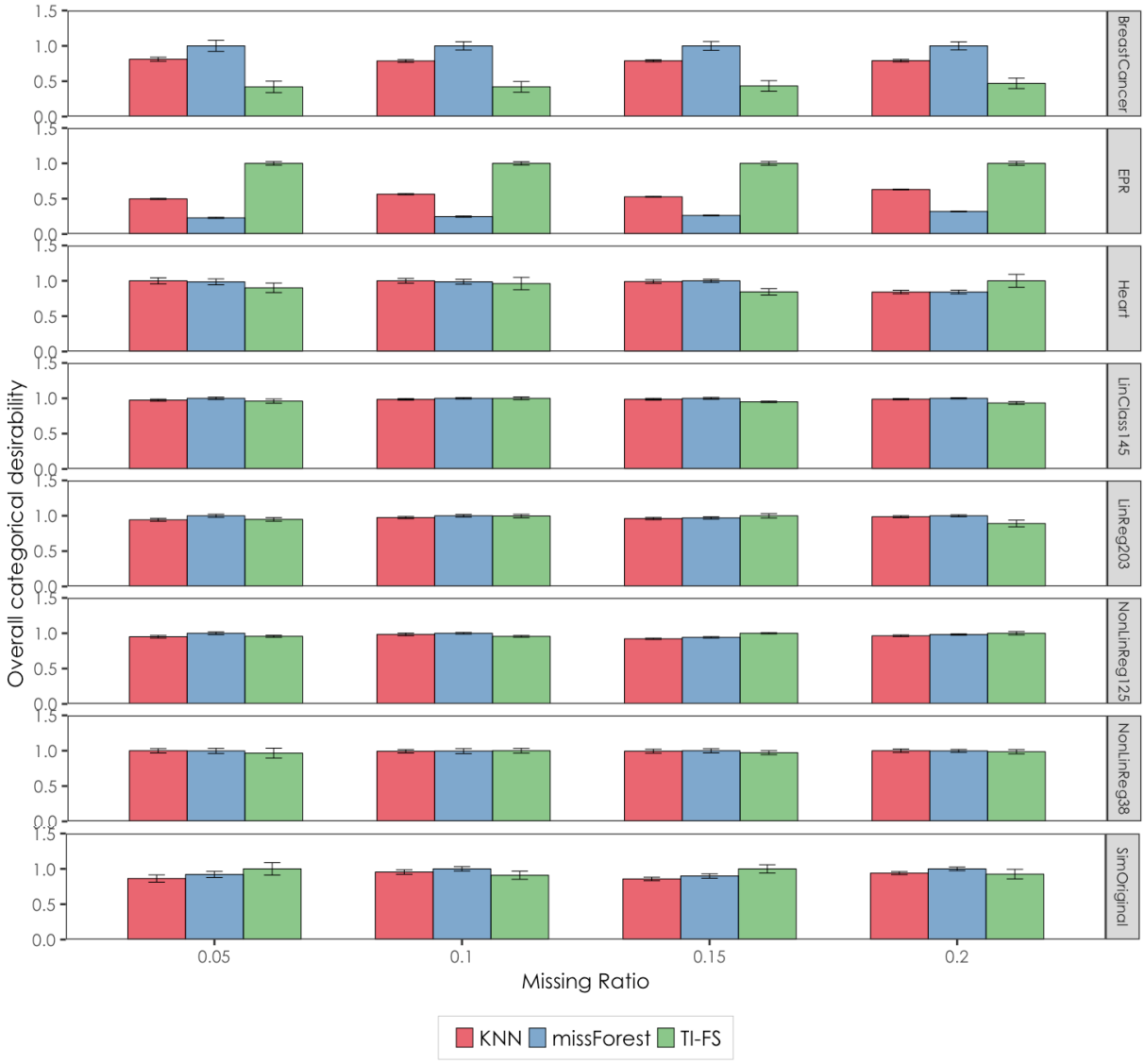


Figure 8: Average performance of the imputation methods based on the categorical desirability. Please note that results are not available for NonLinReg70 data set since it only has numerical variables.

Table 10: Comparison of performance and amount of variables used by TI-FS and missForest.

| Data set | $D'_o$ | | | Used variables | | |
|---|---|---|---|---|---|---|
| | TI-FS | missForest | %change | TI-FS | missForest | %change |
| Heart | 0.8042 | 1.000 | 24.4 | 10 | 13 | 30.0 |
| Breast Cancer | 0.4356 | 1.000 | 129.5 | 7 | 10 | 42.9 |
| EPR | 0.6224 | 1.000 | 60.7 | 70 | 99 | 41.4 |
| LinReg203 | 0.9648 | 1.000 | 3.6 | 86 | 203 | 136.0 |
| LinClass145 | 0.9597 | 0.9940 | 3.6 | 85 | 145 | 70.6 |
| NonLinReg70 | 0.9702 | 0.9986 | 2.9 | 38 | 70 | 84.2 |

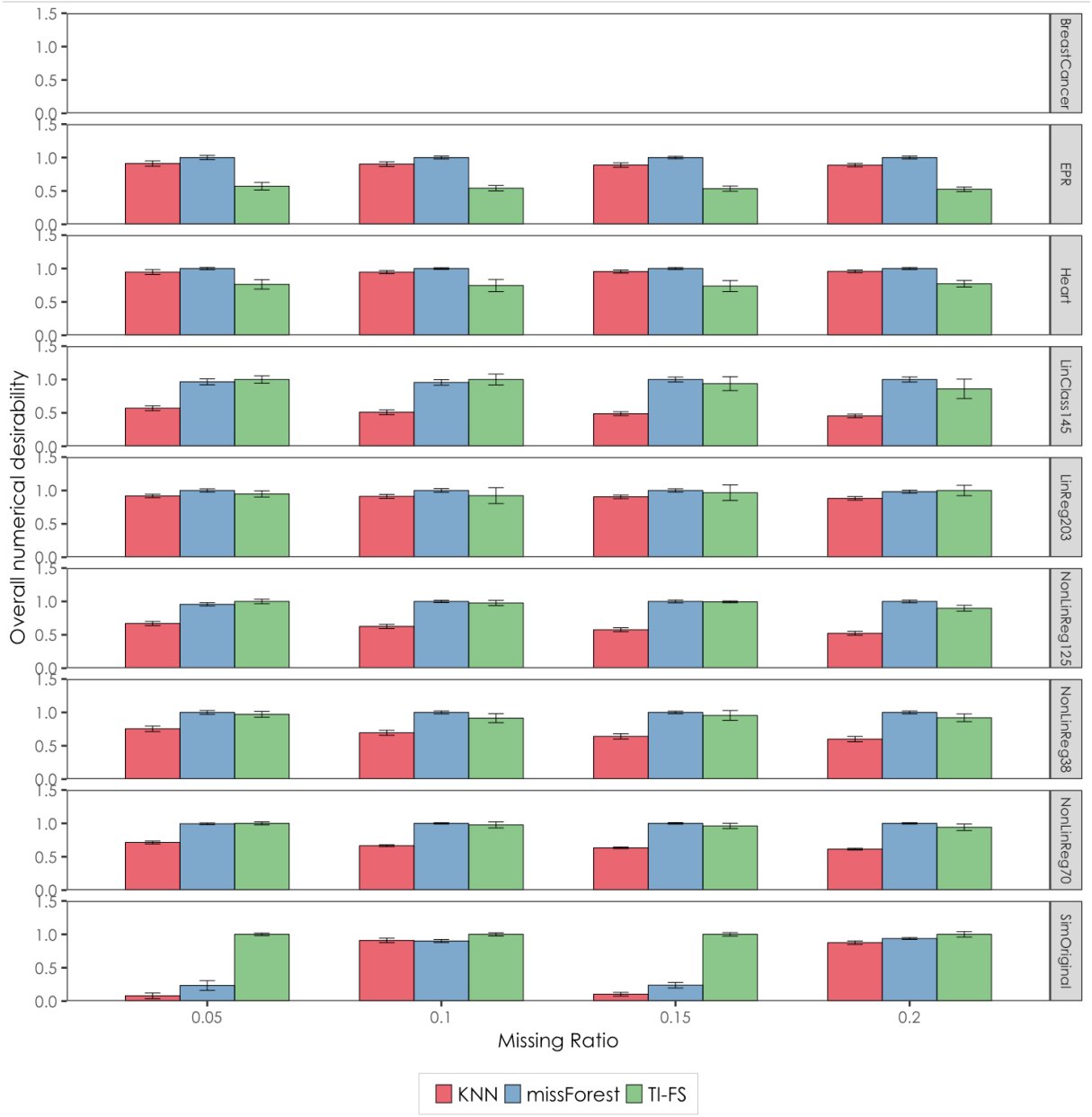


Figure 9: Average performance of the imputation methods based on the numerical desirability. Please note that results are not available for Breast Cancer data set since it only has categorical variables.

## 6 Conclusions

Properly imputed data gives the opportunity to retrieve, not only the best possible predictions for the missing values, but to replace them with reliable values. The goal of any successful MVI scheme is to exploit the information in the incomplete cases and effectively develop approaches to better understand the underlying populations described in these data sets.

In this paper, a tree-based imputation method using FS has been proposed. This method

considers the relationships among variables using GA FS. TI-FS is intended for use in complex data sets with a moderate to large amount of missing values.

Results showed that the proposed method has good performance in scenarios which have complex linear and non-linear relationships between variables. The results also suggested that TI-FS has better performance imputing categorical variables than numerical variables, specially at higher missing ratios, where the percentage difference is no more than 3.6% of missForest.

TI-FS still a reasonable approximation if considered that it had, on average, 60% of the total predictors available to carry out the imputations and, thus, required substantially less preliminary imputations. Another difference is that the convergence criterion considers a global convergence criteria in missForest, while the approach for TI-FS is global convergence criteria. This, in turn, could translate to a scheme that is converges faster than its multivariate counterpart.

As future improvements, it is suggested to evaluate other computationally-feasible imputation for the initial guess of the data (e.g. a RF-based imputation). This has a direct impact on the performance of the FS method and, therefore, in the imputation. In conjunction with better preliminary imputations, improvements on FS can further reduce the computational cost of the MVI scheme and, thus, allow for a more efficient model of the relationship between the variable being imputed and the remaining independent variables in the data set which, in turn, translates to better missing value estimates.

## References

Andridge RR, Little RJA (2010). A review of hot deck imputation for survey non-response. *International Statistical Review*, 78(1): 40–64.

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

Breiman L, Friedman J, Stone CJ, Olshen RA (1984). *Classification and Regression Trees*. Chapman and Hall/CRC.

Brownlee J (2014). An introduction to feature selection. http://machinelearningmastery.com/an-introduction-to-feature-selection.

Burgette L, Reiter J (2010). Multiple imputation for missing data via sequential regression trees. *American Journal of Epidemiology*, 172(9): 1070–1076.

Chandrashekar G, Sahin F (2014). A survey on feature selection methods. *Computers and Electrical Engineering*, 40: 16–28.

Comulada WS (2015). Model specification and bootstrapping for multiply imputed data: An application to count models for the frequency of alcohol use. *The Stata Journal*, 15(3): 833–844.

Dávila S, Rosado H (2017). Performance of missing value imputation schemes in health-related data. In: *IIE Annual Conference Proceedings*, 2105–2110.

Detrano R, Janosi A, Steinbrunn W, Pfisterer M, Schmid JJ, Sandhu S, et al. (1989). International application of a new probability algorithm for the diagnosis of coronary artery disease. *The American Journal of Cardiology*, 64(5): 304–310.

Doshi M, Chaturvedi (2014). Correlstion based feature selection (CFS) technique to predict student performance. *International Journal of Computer Networks & Communications*, 6(3): 197–206.

Enders CK (2010). *Applied Missing Data Analysis*. The Guiford Press, New York, NY, USA.

Friedman J (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, (5): 1189–1232.

Gelman A, Hill J (2006). Missing-data imputation. In: *Data Analysis Using Regression and Multilevel/Hierarchical Models*, 529–544. Cambridge University Press.

Genuer R, Poggi J, Tuleau-Malot C (2015). VSURF: An R package for variable selection using random forests. *The R Journal*, 7(2): 19–33.

Gower JC (1971). A general coefficient of similarity and some of its properties. *Biometrics*, 27(4): 857–871.

Grau J, Keilwagen J (2018). *PRROC: Precision-Recall and ROC Curves for Weighted and Unweighted Data*. R package version 1.3.1.

Guyon I, Elisseeff A (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3: 1157–1182.

Hall MA (2000). Correlation-based feature selection for discrete and numeric class machine learning. In: *Proceedings of the Seventeenth International Conference on Machine Learning*, 359–366. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Kaiser J (2014). Dealing with missing values in data. *Journal of Systems Integration*, 5(1).

Kira K, Rendell L (1992). The feature selection problem: Traditional methods and a new algorithm. *AAAI Proceedings*.

Kohavi R, John G (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97: 273–324.

Kowarik A, Templ M (2016). Imputation with the R package VIM. *Journal of Statistical Software*, 74(7).

Kuhn M (2020). *caret: Classification and Regression Training*. R package version 6.0-86.

Lei Y, Liu H (2003). Feature selection for high-dimensional data: A fast correlation-based filter solution. In: *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA*.

Liao SG, Lin Y, Kang DD, Chandra D, Bon J, Kaminski N, et al. (2014). Missing value imputation in high-dimensional phenomic data: Imputable or not, and how? *BioMed Central Bioinformatics*, 15: 346.

Liaw A, Wiener M (2002). Classification and regression by randomforest. *R News*, 2(3): 18–22.

Lichman M (2013). UCI machine learning repository. http://archive.ics.uci.edu/ml.

Little RJA, Schluchter MD (1985). Maximum likelihood estimation for mixed continuous and categorical data with missing values. *Biometrika*, 72(3): 497–512.

López V (2005). Comparación de los métodos de imputación con respecto al poder de separación del modelo de regresión logistíca. Ph.D. thesis, University of Puerto Rico.

Mohamad M, Deris S, Razib M (2004). Feature selection method using genetic algorithm for the classification of small and high dimension data. *First Internation Symposium on Information and Communications Technologies*.

Pantanowitz A, Marwala T (2009). Missing data imputation through the use of the random forest algorithm. In: *Advances in Computational Intelligence* (J Kacprzyk, W Yu, EN Sanchez, eds.), volume 116, 53–62. Springer Berlin Heidelberg, Berlin, Heidelberg.

Revelle W (2019). *psych: Procedures for Psychological, Psychometric, and Personality Research*. Northwestern University, Evanston, Illinois. R package version 1.9.12.

Robnik-Sikonja M, Kononenko I (2003). Theoretical and empirical analysis of ReliefF and RReliefF. *Machine Learning Journal*, 53(53): 23–69.

Rogier A, Donders T, van der Heijden G, Stijnend T (2006). Review: A gentle introduction to imputation of missing values. *Journal of Clinical Epidemiology*, 59: 1087e1091.

Romanski P, Kotthoff L (2018). *FSelector: Selecting Attributes*. R package version 0.31.

Saeys Y, Inza In, Larrañaga P (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19): 2507–2517.

Schafer JL (1997). *Analysis of Incomplete Multivariate Data*. CRC Press.

Sertkaya A, Birkenbach A, Berlind A, Eyraud J (2014). Examination of clinical trial costs and barriers for drug development. *Technical report*, Department of Health and Human Services.

Shardlow M (2008). An analysis of feature selection techniques. Working Paper.

Silva E, Pino R, Lopez M, Cubiles M (2011). Missing value imputation on missing completely at random data using multilayer perceptrons. *Neural Networks*, 24: 121–129.

Sim J, Lee JS, Kwon O (2015). Missing values and optimal selection of an imputation method and classification algorithm to improve the accuracy of ubiquitous computing applications. *Mathematical Problems in Engineering*, 2015: 538613.

Singh A, Yadav A, Rana A (2013). K-Means with three different distance metrics. *International Journal of Computer Applications*, 67(10): 13–17.

Stekhoven D (2013). missForest: Nonparametric missing value imputation using random forest. R package version 1.4.

Stekhoven DJ, Bülmann P (2012). Missforest-non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1): 112–118.

Sulis I, Porcu M (2008). Assessing the effectiveness of a stochastic regression imputation method for ordered categorical data. Working Paper.

Team RC (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Tuv E, Borisov A, Runger G, Torkkola K (2009). Feature selection with ensembles, artificial variables, and redundancy elimination. *Journal of Machine Learning Research*, 10: 1341–1366.

US Forest Service (2006). SYLVA is an ecology dataset. http://www.causality.inf.ethz.ch/al_data/SYLVA.html.

Wolberg WH, Mangasarian OL (1990). Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences*, 87(23): 9193–9196.

Wood AM, White IR, Thompson SG (2004). Are missing outcome data adequately handled? A review of published randomized controlled trials in major medical journals. *Clinical Trials*, 1(4): 368–376.

Yeşilova A, Kaya Y, Almali N (2010). A comparison of hot deck imputation and substitution methods in the estimation of missing data. *Gazi University Journal of Science*, 24(1): 69–75.

Zambrano-Bigiarini M (2020). *hydroGOF: Goodness-of-Fit Functions for Comparison of Simulated and Observed Hydrological Time Series*. R package version 0.4-0.