Int. J. Appl. Math. Comput. Sci., 2026, Vol. 36, No. 2, 223–234
DOI: 10.61822/amcs-2026-0016

BENCHMARKING IMPUTATION METHODS FOR FUZZY DATASETS

MACIEJ ROMANIUK [a]

[a] Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6, 01-447 Warsaw, Poland
e-mail: mroman@ibspan.waw.pl

Imputation methods are widely used to replace missing values in datasets, thereby improving the overall quality of samples and enabling further statistical procedures. Various measures and tools have been proposed to compare the effectiveness and results of imputation algorithms. This paper describes the extended benchmarking approach designed explicitly for imputing fuzzy datasets. It is intended as a unique combination of classical tools with new measures that address the special features of fuzzy sets. With the help of this benchmark, five imputation methods (the widely known missForest, miceRanger, kNN, and PMM algorithms, as well as the dimp method aimed specifically for fuzzy data) are numerically compared using various synthetic, real-life, single- and multivariate datasets. It is the first such comparison explicitly related to fuzzy data. The obtained conclusions shed new light on the existing, yet still overlooked, problem of imputing missing fuzzy data.

Keywords: missing data, fuzzy sets, random forests, kNN method, numerical comparisons.

1. Introduction

Missing values in datasets pose a significant problem in real-life applications, as they can result in errors, misleading conclusions, or even false predictions (Jadhav *et al.*, 2019; Schafer, 1997). Such a missing value (usually denoted by NA) may also be related to existing but erroneous observations (e.g., a person's negative height). Various mechanisms can lead to the missingness, like data *missing entirely at random* (MCAR), *missing at random* (MAR), or *missing not at random* (MNAR). Therefore, statisticians employ various methods to deal with missing values in datasets, and imputation (i.e., replacing missing data with appropriately selected substitute values) is one of the most important ones (Donders *et al.*, 2006).

There are plenty of imputation approaches (Donders *et al.*, 2006; Little and Rubin, 2022; Dzulkalnine and Sallehuddin, 2019; Sefidian and Daneshpour, 2019), e.g., mean (median or mode) substitution, hot-deck and cold-deck imputation, regression imputation, various clustering methods, bootstrap approaches, algorithms based on ML (machine learning), and fuzzy approaches like fuzzy c-means, fuzzy-rough sets, etc. Ensuring the quality of the imputation method is also necessary. This can be achieved by applying various benchmarks (Jadhav *et al.*, 2019; Jäger *et al.*, 2021; Gendre *et al.*, 2024), which are also part of specialized software (Beck *et al.*, 2018). However, these imputation methods and benchmarks only aim at "crisp" (i.e., real-valued) datasets.

To our best knowledge, only one imputation approach, called *dimp*, exists that is explicitly tailored for fuzzy data, such as triangular and trapezoidal numbers (Romaniuk and Grzegorzewski, 2026). In the fuzzy setting, the problem of missingness is not only limited to the availability of a specific value (as in the case of crisp data) but also related to unique features of fuzzy data (like assumptions concerning membership functions). Moreover, only a "part" of the whole fuzzy value can be missing, e.g., the left end of its support, and the single adequate real value has to be imputed in such a case (see also Section 2.3).

In this paper, we propose an extended set of various benchmarks aimed specifically for checking the quality of imputation methods for fuzzy datasets, including the classical error measures (e.g., MAE, MSE, NRMSE (Jadhav *et al.*, 2019; Beck *et al.*, 2018)), sample (descriptive) statistics (like the means and standard deviations), comparisons of p-values for epistemic goodness-of-fit tests (Grzegorzewski and Romaniuk,

2022; 2024), and distances for the membership functions (based on the Euclidean, AHD (Amirfakhrian *et al.*, 2018), HSD (Yeganehmanesh *et al.*, 2018), and Bertoluzza (Lubiano *et al.*, 2016) metrics), as well as various characteristics of fuzzy data (like width, ambiguity, etc. (Ban *et al.*, 2015) related to the differences between the original and imputed fuzzy data. These benchmarks can be found in the R package `FuzzyImputationTest` (Romaniuk, 2025b). To illustrate the usefulness of the tools considered, we propose the first comparison of different imputation methods for various fuzzy datasets, not their "crisp" counterparts as usual. The imputation algorithms applied include both the "standard" ones (i.e., aimed at "crisp" values) and the single existing method *dimp* designed for the fuzzy data. In the first case, the approaches reported as one of the best (Jäger *et al.*, 2021) are used: *missForest* (Stekhoven and Buehlmann, 2012), *miceRanger* (Royston, 2004), *kNN* (Kowarik and Templ, 2016), and *PMM* (van Buuren and Groothuis-Oudshoorn, 2011). These comparisons are conducted on both synthetic and real-life datasets, including one- and multivariate samples, marking the first known attempt to find the best solution to the imputation problem for fuzzy data. Similar analysis can be repeated using the functions from R packages `FuzzyImputationTest` and `FuzzySimRes` (Romaniuk *et al.*, 2024).

Then, our contribution is threefold. Firstly, we significantly extend the benchmark proposed by Romaniuk and Grzegorzewski (2026) with new tools designed specifically for fuzzy settings, including comparisons based on the Bertoluzza measure, various characteristics of fuzzy data, and methods to assess the errors of these comparisons. Secondly, a detailed numerical analysis (including numerical effectiveness) of various imputation approaches for different types of fuzzy datasets is provided. Thirdly, the results obtained lead us to conclusions that are helpful for users who have to deal with missing values in fuzzy samples.

Please note that the proposed benchmarks are aimed at a fuzzy setting. Some of them (i.e., the classical error measures, like the mean absolute error (MAE), its weighted counterpart (WMAE), the mean squared error (MSE), its weighted counterpart (WMSE), and the normalized root mean squared error (NRMSE) are widely used for real-valued datasets. Hence, with these benchmarks, results similar to those found in our analysis can be obtained without taking into account the specific nature of fuzzy datasets. The same applies to measures based on the means and standard deviations. However, other benchmarks explicitly take into account the discussed fuzziness of data, such as the epistemic goodness-of-fit test, calculation of similarities between membership functions, and the mean differences between specific and valuable characteristics of fuzzy numbers (i.e., the width, ambiguity, expected value, and value). The practically significant percentage of incorrectly generated fuzzy numbers is another example. Hence, these benchmarks are entirely new and allow us to correctly compare the imputation methods for the special case of fuzzy datasets.

The paper is organized as follows. Preliminaries concerning fuzzy numbers and imputation methods are presented in Section 2. The proposed benchmarks are discussed in Section 3. They are then used to compare the imputation algorithms for various fuzzy datasets in Section 4. Finally, concluding remarks are presented in Section 5.

2. Preliminaries

We start by recalling some basic facts concerning fuzzy numbers (see, e.g., Ban *et al.*, 2015). Next, a brief review of some real-valued imputation methods is provided. Some issues concerning fuzzy datasets and their imputation are also discussed.

2.1. Fuzzy numbers: Notation.

Definition 1. (*Fuzzy numbers*) A *fuzzy number* (FN for short) is an imprecise value (Ban *et al.*, 2015) characterized by a mapping $\tilde{A} : \mathbb{R} \to [0,1]$ (a *membership function*), such that its α-cut defined by

$$\tilde{A}_\alpha = \begin{cases} \{x \in \mathbb{R} : \tilde{A}(x) \geqslant \alpha\} & \text{if} \quad \alpha \in (0,1], \\ cl\{x \in \mathbb{R} : \tilde{A}(x) > 0\} & \text{if} \quad \alpha = 0 \end{cases} \tag{1}$$

is a nonempty compact interval for each $\alpha \in [0,1]$. The operator cl in (1) denotes the closure.

Such an FN is uniquely described by its membership function $\tilde{A}(x)$ or, alternatively, a family of α-cuts $\{\tilde{A}_\alpha\}_{\alpha\in[0,1]} = \{(\tilde{A}^L_\alpha, \tilde{A}^R_\alpha)\}_{\alpha\in[0,1]}$, where $\tilde{A}_1 = \mathrm{core}(\tilde{A})$ is the *core*, and $\tilde{A}_0 = \mathrm{supp}(\tilde{A})$ is the *support* of the FN.

Many different membership functions of FNs are known in the literature. One widely used family, the so-called *LR-fuzzy numbers* (Parchami *et al.*, 2024; 2025), is defined by

$$\tilde{A}(x) = \begin{cases} 0 & \text{if} \quad x < a_1, \\ L\left(\frac{x-a_1}{a_2-a_1}\right) & \text{if} \quad a_1 \leqslant x < a_2, \\ 1 & \text{if} \quad a_2 \leqslant x < a_3, \\ R\left(\frac{a_4-x}{a_4-a_3}\right) & \text{if} \quad a_3 \leqslant x < a_4, \\ 0 & \text{if} \quad x \geqslant a_4, \end{cases} \tag{2}$$

where $L, R : [0,1] \to [0,1]$ are continuous and strictly increasing functions such that $L(0) = R(0) = 0$ and $L(1) = R(1) = 1$, and $a_1, a_2, a_3, a_4 \in \mathbb{R}$, where $a_1 \leqslant a_2 \leqslant a_3 \leqslant a_4$. When L and R are linear functions, i.e.,

$L\left(\frac{x-a_1}{a_2-a_1}\right) = \frac{x-a_1}{a_2-a_1}$ and $R\left(\frac{a_4-x}{a_4-a_3}\right) = \frac{a_4-x}{a_4-a_3}$, then we get a *trapezoidal fuzzy number* (TPFN), and if $a_2 = a_3$, we have a *triangular fuzzy number* (TRFN). Each TPFN can be characterized with $\tilde{A} = (a_1, a_2, a_3, a_4)$, and a TRFN—with $\tilde{A} = (a_1, a_2, a_4)$.

TPFNs have many advantages. They are versatile, easy to use and interpret for practitioners, simplify computer applications and approximate FNs having more complex membership functions (Ban *et al.*, 2015).

Usually, FNs are used to model imprecision (e.g., when the results of some experiment cannot be precisely described, qualified, or measured). When there is also an interplay with additional randomness (hence, uncertainty related to the probability theory), we can use *fuzzy random variables* (or *random fuzzy numbers*). Two different viewpoints for such objects, i.e., the *ontic* (Kwakernaak, 1978; Kruse, 1982) and *epistemic* one (Puri and Ralescu, 1986), should be mentioned here.

There are various theoretical attempts concerning the notion of "fuzzy observation", including those based on statistical approaches (see, e.g., Tamaki *et al.*, 1998; Okuda *et al.*, 1990; Hesamian *et al.*, 2023; Calcagnì *et al.*, 2025). However, we focus on the more practically oriented viewpoint in the following. We assume that our datasets are simply sets consisting of some TRFNs/TPFNs with missing values that should be replaced with their "most proper" (in some sense) counterparts. However, because some of our benchmarks are more statistically oriented, we can also think more precisely about samples of fuzzy random variables (see, e.g., Parchami *et al.*, 2024).

2.2. Data imputation methods and their benchmarking. The literature concerning imputation methods, their reviews, and comparisons based on various benchmarks is truly abundant (Jadhav *et al.*, 2019; Donders *et al.*, 2006; Little and Rubin, 2022; Dzulkalnine and Sallehuddin, 2019; Jäger *et al.*, 2021; Gendre *et al.*, 2024; Afkanpour *et al.*, 2024; Lee *et al.*, 2024). To limit the length of the paper, we focus only on the imputation methods applied further. The missForest algorithm (Stekhoven and Buehlmann, 2012) from the R package (Stekhoven, 2022) is an iterative, non-parametric approach based on random forests. These random forests, built on averaging over many unpruned classification or regression trees, result in a multiple imputation scheme. The miceRanger implemented by Wilson (2021) couples *multiple imputation by chained equations* (MICE) (van Buuren and Groothuis-Oudshoorn, 2011) with random forests. The kNN imputation method developed by Templ *et al.* (2021) finds k nearest neighbors for a missing value from all complete samples and then imputes it with the most frequent value occurring in the neighbors or their mean (Zhang, 2012). The PMM (predictive mean matching) from the R package `mice` (van Buuren and Groothuis-Oudshoorn, 2011) calculates the predicted value of the target variable based on a small set of candidate donors. The dimp algorithm (Romaniuk and Grzegorzewski, 2026) is the successor of the resampling method (Romaniuk and Hryniewicz, 2019; 2021) for fuzzy data and uses increments calculated from the dataset to fill the missing values.

2.3. Imputation of fuzzy datasets. Fuzzy datasets related to real-life applications are not as commonly found in the literature as their real-valued counterparts. However, some examples of data can be mentioned here: electronic circuit thickness (essential quality characteristics in the process of producing electronic boards for vacuum cleaners) used to construct a fuzzy statistical control chart (Faraz and Shapiro, 2010) and in the epistemic bootstrap tests (Grzegorzewski and Romaniuk, 2024); testers' opinions concerning various features of the Gamonedo cheese applied in statistical testing of the quality of this cheese (Ramos-Guajardo *et al.*, 2019; Grzegorzewski *et al.*, 2020); performance evaluation of employees used in the fuzzy regression analysis (Gong *et al.*, 2018); tree quality based on subjective expert judgments about various factors originated from a reforestation project in Spain applied in statistical testing (Colubi, 2009); cognitive response times of the crew of a nuclear power plant control room to an abnormal event used in the fuzzy linear regression model (Kim and Bishu, 1998); data from the TIMSS–PIRLS 2011 questionnaire on reading, mathematics and science applied in testing hypotheses about the means (Lubiano *et al.*, 2016) and simulation studies related to fuzzy medoids (Sinova *et al.*, 2025); meteorological variables collected in some cities in Turkey and used for estimation of correlations (Calcagnì, 2024). These datasets are typically small to medium in size (containing up to 100–200 samples) and consist of TRFNs or TPFNs. Some of them are utilized in our comparisons in Section 4.

Missing data in fuzzy sets are related to two types of issues. Sources for the first one are similar to those for the "crisp" datasets, i.e., missing questionnaires, data omitted during entry, software errors, etc. Then, for example, the weight of some person can be simply missing or described by some negative value. However, the second type of missing data can be attributed to their "fuzzy nature". Fuzzy observation (even in the form of a TRFN or TPFN) is not a simple single value but an "entity" with its inner structure resulting from the existence of a membership function (and the respective requirements given by Definition 1). In the case of a TRFN, such an observation is a triplet (or a quadruplet for a TPFN) satisfying the obvious requirement $a_1 \leq a_2 \leq a_4$ (or

$a_1 \le a_2 \le a_3 \le a_4$, respectively). Hence, it constitutes a vector with some embedded boundary conditions. Omitting even one of its elements leads to the necessity for proper imputation that fulfills these requirements, e.g., we have $\tilde{A} = (30, 46, 44, 54)$ in Table 2 in the work of Ramos-Guajardo *et al.* (2019). This incorrect FN should be replaced (rather than straightforwardly removed) with one of the following: $(30, 46, \text{NA}, 54)$, $(30, \text{NA}, \text{NA}, 54)$, $(30, 46, \text{NA}, \text{NA})$. Next, these missing parts of $\tilde{A}$ should be imputed as "best as possible", so some benchmarks are necessary.

The "classical" benchmarks (like the MAE, NRMSE, etc.; cf. Section 3.1) are also useful for fuzzy datasets. However, they are intended only for real-valued data, not for FNs, which raises the issue of the inner structure mentioned above. Once again, a whole FN (e.g., the entire vector) should be considered to measure the imputation quality. For example, two FNs can be regarded as "similar" if their membership functions are close to each other as indicated by some distance measure. Therefore, such measures are our point of interest (see Section 3.4). The same applies to descriptive statistics when these "classical" benchmarks (e.g., the mean; cf. Section 3.2) are entirely valid, but they check the imputation quality only "point-to-point" (not for the whole FN). Introduction of additional benchmarks based on the "statistical fuzzy nature" of datasets (like epistemic goodness-of-fit tests, explicitly intended for FNs; cf. Section 3.3) is then necessary to compare the samples consisting of FNs.

Moreover, a special imputation method designed for datasets consisting of TRFNs or TPFNs may also be valuable, like the dimp algorithm (Romaniuk and Grzegorzewski, 2026). FNs from the input sample are decomposed into three (in the case of TRFNs) or four (for TPFNs) special sets during the initialization step of this method. These sets contain the cores (in the case of TRFNs) or left endpoints and increments of the cores (for TPFNs), left increments of the supports, and their right counterparts. Next, the missing values in the sample are randomly replaced by the values from these constructed sets. In the case of an FN showing some missingness, the algorithm starts from its core (if this core contains some NAs, e.g., the left endpoint of a TPFN is missing) and then proceeds to its support (if there are some NAs there, e.g., the right endpoint of the support does not exist). This method is applied as one of the imputation algorithms in `FuzzyImputationTest`.

3. Benchmarking measures

The benchmarks considered represent an exceptional combination of classical approaches (previously used only for "crisp" datasets) and entirely new ideas tailored to the unique features of fuzzy data. They can be divided into five groups (see Sections 3.1–3.5). To properly distinguish between the different methods (in the sense of these benchmarks), we also calculate the respective standard errors (SEs).

We assume we have a dataset with a single fuzzy variable X to simplify the notation. Then, let $\mathbb{X} = (X_1, \ldots, X_n)$ be a complete sample (i.e., without any missingness) of n observations given by TRFNs (or TPFNs). Such an $\mathbb{X}$ is identical to a real-valued matrix of size $n \times r$ (where $r = 3$ for TRFNs or $r = 4$ for TPFNs) for which each row $X_i = (x_{i,1}, \ldots, x_{i,r})$ describes the observed FN X_i.

Two additional matrices are then introduced. The first one, $\mathbb{X}^*$, consists of the original values from $\mathbb{X}$ and some NAs (i.e., *no value is available*). These NAs are introduced independently for each column of $\mathbb{X}$ when np_{imp} random cells are selected in each column and their initial content is replaced with NAs (resulting in the MCAR setup). The parameter $p_{\text{imp}} \in (0, 1)$ is the *missing values percentage*. Then, some FNs from $\mathbb{X}^*$ may lack one or even more of their focal points (e.g., the whole support). The matrix $\mathbb{X}^*$ can be reorganized into two submatrices: $\mathbb{X}^{\text{NA}}$ of the size $s \times r$ with at least a single NA in some row, and $\mathbb{X}^{\text{no-NA}}$ of the size $(n - s) \times r$ with the completely known all values. We can rewrite $\mathbb{X}$ as $\mathbb{X} = \left(\mathbb{X}^{\text{true}}, \mathbb{X}^{\text{no-NA}}\right)$, where the submatrix $\mathbb{X}^{\text{true}} = (X_1^{\text{true}}, \ldots, X_s^{\text{true}})$ contains a true (but then masked with NAs in $\mathbb{X}^{\text{NA}}$) values.

The second matrix, $\mathbb{Y} = \left(\mathbb{Y}^{\text{imp}}, \mathbb{X}^{\text{no-NA}}\right)$, is created after the imputation procedure, where $\mathbb{Y}^{\text{imp}} = \left(Y_1^{\text{imp}}, \ldots, Y_s^{\text{imp}}\right)$ of the size $s \times r$ contains s imputed TRFNs (or TPFNs, respectively) replacing the NAs previously existing in $\mathbb{X}^{\text{NA}}$.

3.1. Classical error measures. The first obvious quality measure is the number of the imputed values that do not fulfill the obvious assumptions $y_{i,1}^{\text{imp}} \le y_{i,2}^{\text{imp}} \le y_{i,3}^{\text{imp}}$ (for TRFNs) or $y_{i,1}^{\text{imp}} \le y_{i,2}^{\text{imp}} \le y_{i,3}^{\text{imp}} \le y_{i,4}^{\text{imp}}$ (for TPFNs), referred to as *incorrect FNs*.

Let us consider other classical error measures for real-valued datasets (Jadhav *et al.*, 2019; Beck *et al.*, 2018). The *mean absolute error*,

$$\begin{aligned} \text{MAE} &= \frac{1}{r}\sum_{j=1}^{r} \text{MAE}_j, \\ \text{MAE}_j &= \frac{1}{np_{\text{imp}}}\sum_{i=1}^{np_{\text{imp}}} \left| x_{i,j}^{\text{true}} - y_{i,j}^{\text{imp}} \right|, \end{aligned} \tag{3}$$

is calculated (point-wise) only between the true values $x_{i,j}^{\text{true}}$ and their imputed counterparts $y_{i,j}^{\text{imp}}$ for $\mathbb{X}^{\text{true}}$ and $\mathbb{Y}^{\text{imp}}$, respectively. Hence, MAE_j is the error for the j-th column, and the MAE is the overall mean value for all

MAEs. We also have the *weighted mean absolute error*,

$$\mathrm{WMAE}_j = \frac{1}{np_{\mathrm{imp}}} \sum_{i=1}^{np_{\mathrm{imp}}} \frac{\left| x_{i,j}^{\mathrm{true}} - y_{i,j}^{\mathrm{imp}} \right|}{\left| x_{i,j}^{\mathrm{true}} \right|}, \tag{4}$$

where the denominator is set to one when $x_{i,j}^{\mathrm{true}} = 0$ and the WMAE is the respective overall mean of WMAEs for all columns. Similarly, we calculate the *mean squared error* and *weighted mean squared error*,

$$\mathrm{MSE}_j = \frac{1}{np_{\mathrm{imp}}} \sum_{i=1}^{np_{\mathrm{imp}}} \left(x_{i,j}^{\mathrm{true}} - y_{i,j}^{\mathrm{imp}} \right)^2, \tag{5}$$

$$\mathrm{WMSE}_j = \frac{1}{np_{\mathrm{imp}}} \sum_{i=1}^{np_{\mathrm{imp}}} \frac{\left(x_{i,j}^{\mathrm{true}} - y_{i,j}^{\mathrm{imp}} \right)^2}{\left(x_{i,j}^{\mathrm{true}} \right)^2}, \tag{6}$$

with the respective MSE and WMSE counterparts. The *normalized root mean squared error* is defined by

$$\mathrm{NRMSE}_j = \sqrt{\frac{1}{np_{\mathrm{imp}}} \sum_{i=1}^{np_{\mathrm{imp}}} \frac{\left(x_{i,j}^{\mathrm{true}} - y_{i,j}^{\mathrm{imp}} \right)^2}{\max\limits_{k\in\{1,\ldots,n\}}(x_{k,j}) - \min\limits_{k\in\{1,\ldots,n\}}(x_{k,j})}}, \tag{7}$$

where the maximum and minimum are taken regarding each j-th column of $\mathbb{X}$.

3.2. Descriptive statistics. The actual values can also be compared with their imputed counterparts using descriptive statistics, such as the mean or standard deviation. Let us define

$$\bar{x}_j^{\mathrm{true}} = \frac{1}{np_{\mathrm{imp}}} \sum_{i=1}^{np_{\mathrm{imp}}} x_{i,j}^{\mathrm{true}}, \bar{y}_j^{\mathrm{imp}} = \frac{1}{np_{\mathrm{imp}}} \sum_{i=1}^{np_{\mathrm{imp}}} y_{i,j}^{\mathrm{imp}}, \tag{8}$$

so $\bar{x}_j^{\mathrm{true}}$ is the mean of the true values (but then replaced with NAs) for the j-th column of $\mathbb{X}^{\mathrm{true}}$, and $\bar{y}_j^{\mathrm{imp}}$ is the mean of their imputed counterparts for the j-th column of $\mathbb{Y}^{\mathrm{imp}}$. The usual means for the j-th columns of $\mathbb{X}$ and $\mathbb{Y}$ are given by

$$\bar{x}_j = \frac{1}{n} \sum_{i=1}^{n} x_{i,j}, \bar{y}_j = \frac{1}{n} \sum_{i=1}^{n} y_{i,j}. \tag{9}$$

The standard deviations $s_j^{x,\mathrm{true}}, s_j^{y,\mathrm{imp}}, s_j^x, s_j^y$ can be calculated similarly to (8)–(9). If the imputation procedure "mimics" the statistical properties of the actual data correctly, all these measures should be close to each other for $\mathbb{X}$ and $\mathbb{Y}$, i.e., for each j, $\bar{x}_j^{\mathrm{true}}$ should be similar to $\bar{y}_j^{\mathrm{imp}}$, $\bar{x}_j$ to $\bar{y}_j$, etc., and their absolute differences should be close to zero.

3.3. Epistemic goodness-of-fit tests. Another way to assess the quality of the imputation is to apply a goodness-of-fit test. Due to the fuzzy nature of our datasets, we employ the epistemic approach for the two-sample Kolmogorov–Smirnov (EKS for short) test, specifically its antithetic variant, which is known for its good statistical properties (Grzegorzewski and Romaniuk, 2022; 2024). With the help of the package `FuzzySimRes` (Romaniuk *et al.*, 2024), the following p-values can be calculated (see also Romaniuk and Grzegorzewski, 2026): $p_{\text{no-NA,true}}$, when the EKS test compares the true non-masked and masked FNs (i.e., $\mathbb{X}^{\text{no-NA}}$ with $\mathbb{X}^{\text{true}}$); $p_{\text{no-NA,imp}}$, for comparison of the initial non-masked FNs with the imputed values ($\mathbb{X}^{\text{no-NA}}$ with $\mathbb{Y}^{\text{imp}}$); $p_{\text{true,imp}}$, in the case of the original, masked FNs versus their imputed replacements ($\mathbb{X}^{\text{true}}$ versus $\mathbb{Y}^{\text{imp}}$).

The masked FNs and their replacements are statistically identical for the perfect imputation method. Therefore, to verify the quality of the selected algorithm, the difference between $p_{\text{no-NA,true}}$ and $p_{\text{no-NA,imp}}$ (the smaller one, the better) should be calculated. In contrast, $p_{\text{true,imp}}$ should be ideally as close as possible to one.

3.4. Membership functions. Once again, the fuzzy nature of our dataset should be taken into account. Therefore, the similarity between membership functions for the true FNs and their imputed counterparts can be considered with the help of

$$\bar{d}(\mathbb{X}^{\mathrm{true}}, \mathbb{Y}^{\mathrm{imp}}) = \frac{1}{s} \sum_{i=1}^{s} d\left(X_i^{\mathrm{true}}, Y_i^{\mathrm{imp}} \right), \tag{10}$$

where $d(.,.)$ is some distance between two FNs. Various functions $d(.,..)$ can be applied for (10), and we consider the following: the widely used Euclidean metric (Grzegorzewski, 1998), AHD (area height distance) (Amirfakhrian *et al.*, 2018), HSD (height source distance) (Yeganehmanesh *et al.*, 2018), and Bertoluzza measure (Lubiano *et al.*, 2016) (aka mid-spread distance). The smaller values of $\bar{d}(\mathbb{X}^{\mathrm{true}}, \mathbb{Y}^{\mathrm{imp}})$ indicate more significant similarities between the respective membership functions and the better quality of the imputation procedure.

3.5. Fuzzy characteristics. Our set of benchmarks uses mean differences for other widely known characteristics of FNs, i.e., the width, ambiguity, expected value, and value (Ban *et al.*, 2015). For example, the mean difference for the ambiguity is given by

$$\bar{\Delta}_{\mathrm{Amb}}(\mathbb{X}^{\mathrm{true}}, \mathbb{Y}^{\mathrm{imp}}) = \frac{1}{s} \sum_{i=1}^{s} \left| \mathrm{Amb}\left(X_i^{\mathrm{true}} \right) - \mathrm{Amb}\left(Y_i^{\mathrm{imp}} \right) \right|, \tag{11}$$

Table 1. Models of the simulated synthetic fuzzy datasets.

Model	m	n	Type	X	C^l, C^r	S^l	S^r
$\mathbb{F}_{(\mathrm{N,E,U})}$	1	50,200,500	TPFN	$\mathrm{N}(0,1)$	$\mathrm{Exp}(1)$	$\mathrm{U}(0,1)$	$\mathrm{U}(0,1)$
$\mathbb{F}_{(\mathrm{N,U})}$	1	50,200,500	TRFN	$\mathrm{N}(0,1)$	–	$\mathrm{U}(0,1)$	$\mathrm{U}(0,1)$
$\mathbb{F}_{(\Gamma,\mathrm{U},\beta)}$	2	50,200,500	TPFN	$\Gamma(2,4)$	$\mathrm{U}(1,2)$	$\beta(5,1)$	$\beta(2,5)$

where Amb denotes the ambiguity of the respective FN $\tilde{A}$ calculated as

$$\mathrm{Amb}\left(\tilde{A}\right) = \int_0^1 \left(\tilde{A}_\alpha^R - \tilde{A}_\alpha^L\right) d\alpha. \tag{12}$$

Formulas similar to (11) are then used to calculate the mean differences for the width, expected value, and value. When these values are smaller, the imputed FNs can be regarded as closer to their actual counterparts.

3.6. Remarks about the informative value of the benchmarks. We conclude the description of our benchmarks by highlighting their specific and informative values (this can help choose "the most useful benchmark" for each case study):

- *overall real-valued similarity:* the MAE, WMAE, MSE, WMSE (see Section 3.2);
- *general statistical similarity:* the mean differences for the mean, etc. (mainly; see Section 3.2), the EKS test (additionally; see Section 3.3);
- *overall similarity of FNs characteristics:* the mean differences for the FNs characteristics (Section 3.5);
- *preservation of distribution properties:* the EKS test (mainly; see Section 3.3), the mean differences for the mean, etc. (additionally; see Section 3.2);
- *general correctness of the imputed FNs:* the number of the incorrect FNs (mainly Section 3.1), the mean differences for the FNs characteristics (additionally; see Section 3.5);
- *similarity of the membership functions:* distances between the membership functions (mainly Section 3.4), the mean differences for the FNs characteristics (additionally; see Section 3.5).

4. Numerical comparisons

In our analysis, we focus on datasets consisting of TRFNs/TPFNs. The real-valued imputation methods considered can be easily adapted for the matrix-like structure of such data. Both the synthetic (Jordon *et al.*, 2022) and real-life datasets, single- and multivariate, were studied, each consisting of m variables and n samples. All numerical experiments were repeated 500 times to reduce the randomness of the results. We analyzed the influence of the increasing percentage $p_{\text{imp}} \in \{0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4\}$ of the missing values on the quality of the imputation methods measured with the benchmarks introduced in Section 3. We used $p_{\text{imp}} \in \{0.15, 0.2, 0.25, 0.3, 0.35, 0.4\}$ only in the case of the *Performance* dataset (see Section 4.1) because of its small sample size.

To limit the length of the paper, only some examples of results are presented in the following. Additional graphs are in the supplementary file (Romaniuk, 2025a). Detailed results and datasets are available upon reasonable request.

4.1. Employed datasets. The synthetic samples were generated using the parameters described in Table 1 with the help of `FuzzySimRes`. As explained by Grzegorzewski and Romaniuk (2024), Romaniuk *et al.* (2024), as well as Romaniuk and Grzegorzewski (2023), to simulate a TPFN, five independent real-valued random variables can be used: X for its "true value" (i.e., its *original*), C^l, C^r for the left and right increments of the core, and S^l, S^r for the left and right increments of the support. In Table 1, the normal distribution with the mean μ and standard deviation σ is denoted by $\mathrm{N}(\mu, \sigma)$, uniform distribution on the interval (a, b) by $\mathrm{U}(a, b)$, exponential distribution with the parameter λ by $\mathrm{Exp}(\lambda)$, gamma distribution with the shape a and scale b parameters by $\Gamma(a, b)$, and beta distribution with the shape parameters a, b by $\beta(a, b)$. Relatively small (with $n = 50$ elements), moderate ($n = 200$), and rather big ($n = 500$) sample sizes were considered. For $\mathbb{F}_{(\Gamma,\mathrm{U},\beta)}$, the dependency between two fuzzy variables was modeled by adding small noise (i.e., *idd* random draws from $N(0, 1)$) to each focal point of values of the first variable.

The fuzzy real-life datasets (see also Section 2.3) are summarized in Table 2. *ControlChartData* (available in `FuzzySimRes`) contains data (collected and described by Faraz and Shapiro (2010)) related to the control charts for electronic circuit thickness. *GamonedoCheese* is the dataset (taken from the work of Ramos-Guajardo *et al.* (2019)) that contains experts' opinions concerning the overall impressions of the Gamonedo cheese—a kind of blue cheese produced in Asturias, Spain. *Performance* (considered by Gong *et al.* (2018)) includes subjective factors that influence work performance and the response variable. The subsequent two datasets

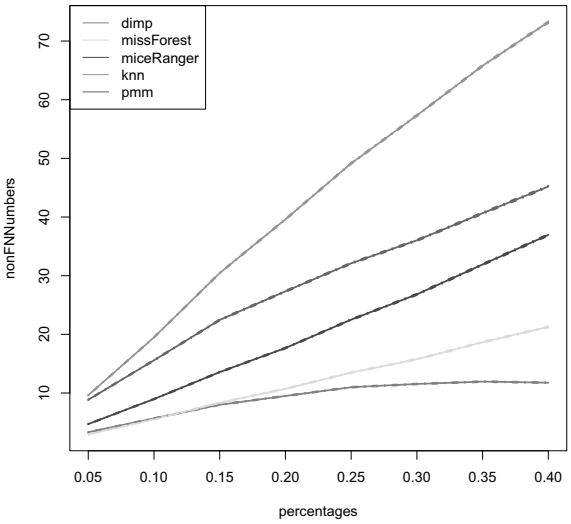

Fig. 1. Generated number of the incorrect FNs for the *IrisFuzzy* dataset.

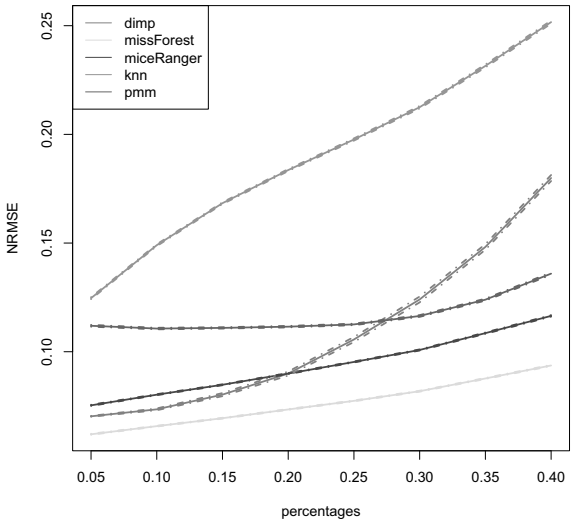

Fig. 2. Estimated NRMSE for the *IrisFuzzy* dataset.

are fuzzified (Kayacan and Khanesar, 2016) versions of "crisp" data with the non-numerical (i.e., categorical) variables omitted. The fuzzification algorithm applied is based on random draws from the uniform distributions parametrized by the standard deviations of each variable and is performed with `FuzzyImputationTest`. Then, *IrisFuzzy* is a fuzzified version of the famous *iris* dataset (Fisher, 1936), containing the measurements of flowers of three species of iris. *WineFuzzy* is based on the *wine* dataset (taken from the `rattle` package (Williams, 2011), considered by Asunción and Newman (2007)) with the results of a chemical analysis of wines grown in a specific area of Italy.

4.2. Results. Examples of graphs of some benchmarks as functions of the increasing p_{imp} for *IrisFuzzy* can be found in Figs. 1–6. The plots for all benchmarks and datasets are included in the supplementary file mentioned earlier. The conclusions and remarks provided in the following are based on all datasets. Apart from the estimators of the respective benchmarks, we also calculated their standard errors (SEs) to assess the statistical significance of the results and accurately identify the best/worst algorithms. These SEs are shown as the respective dot-dashed bounds in all plots.

The results are somewhat complex and varied, yet they lead to some overall conclusions. There seems to be no single winner for all the datasets and benchmarks considered. But missForest usually results in the best quality of the imputed values, i.e., the lowest number of the generated incorrect FNs, values of the MAE/WMAE/MSE/WMSE/NRMSE errors, distances between the membership functions of FNs and their fuzzy characteristics, and differences between the means and standard deviations, together with the highest p-values for the EKS tests. On the contrary, the kNN method leads to many of the worst results (e.g., when *WineFuzzy* is considered) and at most is the "second/third one". It gives noticeably stable low p-values in the EKS test results for all benchmarks and the datasets considered. Considering the estimated SEs and the "rule of thumb", the differences in quality between missForest and kNN are clearly visible. It seems that miceRanger is always in the middle of the pack. It neither leads to the best outcomes nor the worst ones. Usually, PMM performs similarly to, or worse than, the miceRanger algorithm. Sometimes, the benchmarks for PMM and miceRanger are almost indistinguishable regarding their SEs. Still, for others (e.g., moderate or big sample for $\mathbb{F}_{(\Gamma,\mathrm{U},\beta)}$), the differences that are against PMM are clearly visible. Moreover, PMM reports many warnings concerning collinearity, especially for $\mathbb{F}_{(\Gamma,\mathrm{U},\beta)}$ and *Performance* datasets. The dimp approach shows the varying quality. In some cases (e.g., *WineFuzzy* and *Performance*, especially for small values of p_{imp}), this method is the best one, but for other examples (e.g., the small sample for $\mathbb{F}_{(\mathrm{N},\mathrm{E},\mathrm{U})}$), it leads to the worst results. However, it usually generates imputed values with the highest similarity to the original (masked) ones, as indicated by the EKS tests (i.e., their estimated p-values). The lowest differences between the imputed and original values can also be observed when considering the standard deviations. Additionally, the dimp method works rather well in the multivariate setting (like *IrisFuzzy, WineFuzzy*)—it gives the best results or is close to those of missForest.

Some imputed FNs do not fulfill the necessary

Table 2. Real-life datasets.

Name	m	n	Type	Description
ControlChartData	1	90	TRFN	Electronic circuit thickness (Faraz and Shapiro, 2010)
GamonedoCheese	1	118	TPFN	Experts' opinions about cheese (Ramos-Guajardo *et al.*, 2019)
Performance	5	30	TRFN	Performance evaluation (Gong *et al.*, 2018)
IrisFuzzy	4	150	TPFN	Fuzzified version of the *iris* dataset (Fisher, 1936)
WineFuzzy	13	178	TRFN	Fuzzified version of the *wine* dataset (Williams, 2011)

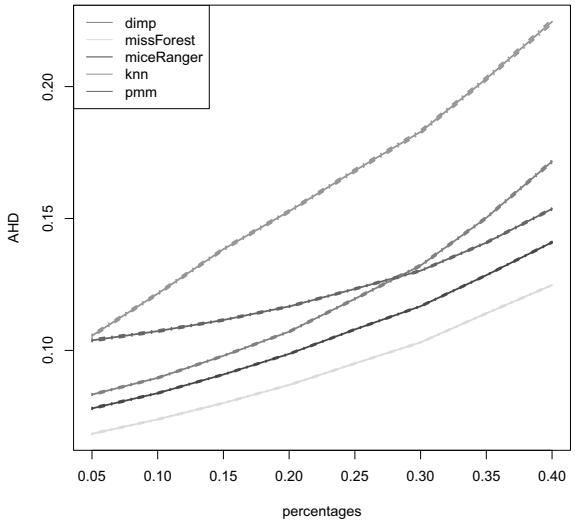

Fig. 3. Estimated AHD distance for the *IrisFuzzy* dataset.

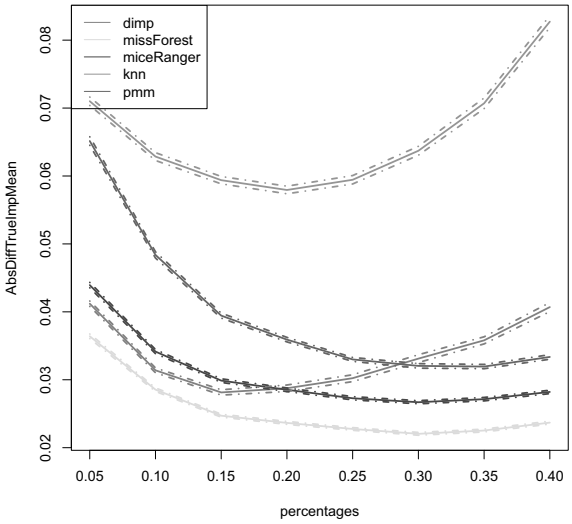

Fig. 4. Absolute difference between the means of the true (masked with NAs) and imputed values for the *IrisFuzzy* dataset.

assumptions, i.e., the obtained value of the incorrect FNs measure is bigger than zero. Then, rejecting such FNs and repeating the imputation procedure for the updated set may be necessary. From our experiments, only dimp requires a small number of such iterations. Other methods may lead to an infinite loop (or too many iterations in practical situations).

When the execution time is considered, our numerical analysis shows that dimp is the fastest method, PMM and kNN occupy the second place (but sometimes PMM is a little faster), miceRanger is the most consuming time algorithm and missForest behaves in a more complex way—sometimes it is close to the second place (i.e., comparable with the PMM/kNN methods), but in other cases (for larger datasets) it falls to the third place because it is significantly slower. The ML imputation methods consume even more time and can give worse results (Jäger *et al.*, 2021; Romaniuk and Grzegorzewski, 2026). Examples of timings obtained with the package `microbenchmark` for 100 repetitions are shown in Figs. 7 and 8. Similar graphs for other datasets and values of p_{imp} can be found in the supplementary file (Romaniuk, 2025a).

Then, it seems that the answer to the question "Which method should be used to impute a fuzzy dataset?" is neither simple nor direct. However, based on our numerical results, the kNN or PMM algorithms are not a good idea. The former usually leads to many of the worst results, together with a lot of low p-values for the EKS goodness-of-fit test, indicating possible problems with the imputed FNs. Similarly, the latter gives noticeable big errors when compared with other methods and generates many warnings about collinearity. The only advantage of these two algorithms is their fast execution times (but dimp is even quicker). Instead, missForest should be used primarily for data with only one variable, and dimp or missForest is advised for multivariate cases. The missForest method usually results in the lowest errors (apart from the number of the incorrect FNs). Its numerical efficiency is not the best one, but it can be similar to the PMM/kNN approaches or sometimes worse. The dimp algorithm behaves in a more complex manner but is very fast, resulting in imputed values of good quality, especially in the multivariate setting. Moreover, dimp seems especially useful when all imputed FNs must be correct, or we are mainly interested in the benchmarks related to the goodness-of-fit EKS tests, or the numerically fastest method is necessary.

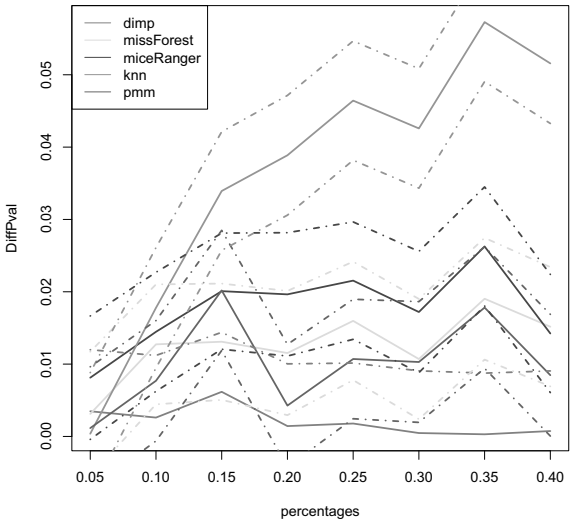

Fig. 5. Estimated difference between $p_{\text{no-NA,true}}$ and $p_{\text{no-NA,imp}}$ in the EKS test for the *IrisFuzzy* dataset.

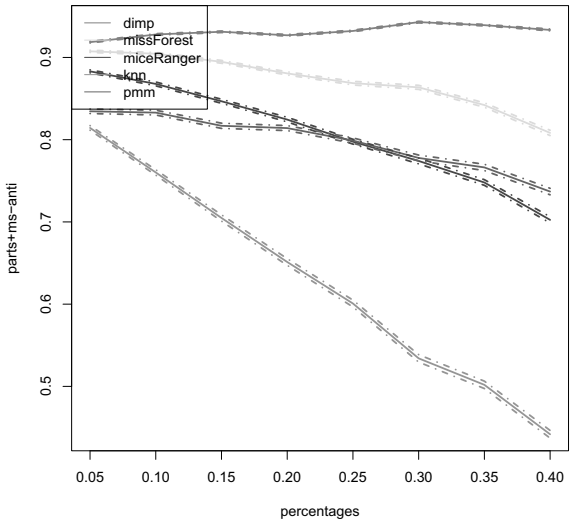

Fig. 6. Estimated $p_{\text{true,imp}}$ in the EKS test for the *IrisFuzzy* dataset.

5. Conclusions

This paper discussed a unique, extended set of benchmarks (contrary to the work of Romaniuk and Grzegorzewski (2026)) for comparing imputation methods. This set was designed explicitly for fuzzy datasets. Because FNs have specific, unique properties (e.g., resulting from their membership functions), apart from more classical tools existing in the literature (like the MAE/MSE/NRMSE), benchmarks related to the descriptive statistics (like the mean), new kinds of measures (e.g., the distances between FNs, p-values of EKS tests designed for FNs) were also considered. Moreover, this is the first attempt to compare the quality of the "crisp" and fuzzy-oriented imputation procedures in the fuzzy setting. Based on our benchmarks, a first numerical analysis for five imputation methods (the widely known kNN, missForest, miceRanger, PMM algorithms, and the new dimp approach designed for FNs) was conducted for different synthetic, real-life, single- and multivariate fuzzy datasets. We focused on commonly used algorithms that are available as R packages and have well-established good properties.

From our experiments, it seems that missForest should be used primarily for data consisting of only one variable, while dimp and missForest are advised for multivariate cases. The two other algorithms (i.e., PMM/kNN) show the overall bad quality of the imputed values (like the high percentage of the incorrect FNs, big values of the errors, low p-values for the goodness-of-fit test, etc.). Their numerical efficiency is satisfactory, but not the best. On the contrary, missForest usually results in the best quality of imputed values, but it may sometimes be slower than PMM/kNN. The miceRanger method is somewhere "in the middle of the pack" (i.e., neither the best nor the worst approach), but it is the most time-consuming algorithm, which is a serious disadvantage. The behavior of dimp is more complex—sometimes it is the best algorithm concerning our benchmarks, and in some cases it is the worst one. Therefore, it should be used with some caution. However, dimp is the fastest method and leads to the lowest percentage of the imputed incorrect FNs. Therefore, it is advisable to consider these features when we are interested in them. When the overall quality of the results is our primary aim, missForest is preferred, despite its lower numerical efficiency.

Regarding the applicability of the introduced methods, two points should be noted. The first one concerns the R package `FuzzyImputationTest` (Romaniuk, 2025b), which provides a set of ready-to-use functions that incorporate the benchmarks introduced in this paper. All the numerical results presented in Section 4 were obtained with the help of this package. The datasets considered were both synthetic and real-life ones. Therefore, similar analyses can be performed for new fuzzy-valued datasets (or easily replicated for the cases mentioned in Section 4). The second issue is related to the necessity of data imputation. As mentioned in Sections 1 and 2.3, datasets with missing values are commonly spotted. Without the proper procedures, the existing lack of information for such samples is a significant problem. In this paper, we addressed this issue through imputation. Hence, we attempted to find an algorithm that is easy to apply for practitioners (e.g., the R package) and numerically efficient, and yields high-quality results for fuzzy-valued datasets. It seems that the "only one of the best" algorithm does not exist, but at least two of them

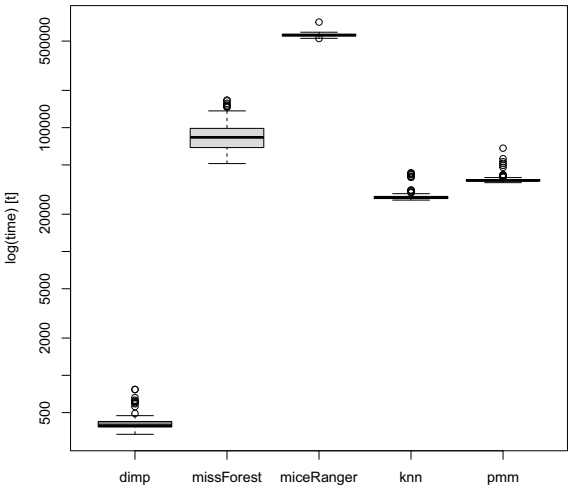

Fig. 7. Timings of imputation methods for $\mathbb{F}_{(\mathrm{N,E,U})}, n = 50$ with $p_{\mathrm{imp}} = 0.2$.

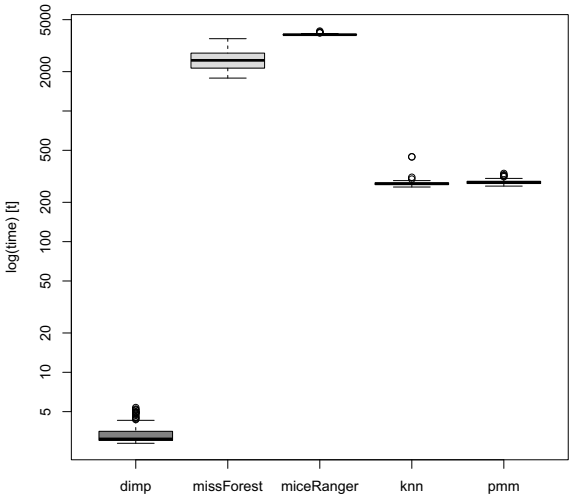

Fig. 8. Timings of imputation methods for *IrisFuzzy* with $p_{\mathrm{imp}} = 0.2$.

(missForest and dimp) can be advised to be applied for new cases.

Of course, there are some limits to our benchmarks and the conducted numerical analysis. We focused on MCAR and imputation of TRNFs/TPFNs. The MCAR mechanism considered is related to the procedures available in `FuzzyImputationTest`. Other missingness patterns require more advanced theoretical mechanisms for handling missing data, as well as the availability of the relevant data (especially fuzzy datasets with many variables). On the other hand, TRFNs/TPFNs were considered because they are families of FNs that are widely used in the literature and are easy to code as real-valued triplets or quadruplets (allowing the existing real-valued imputation methods to be directly applied to them). The respective datasets can be easily found. Another issue is related to computational scalability for very high-dimensional fuzzy data. We analyzed up to 13 variables simultaneously due to existing problems with finding publicly available datasets containing numerous fuzzy-valued variables. Regarding the introduced benchmarks, they can be applied even to multivariate data, albeit with some loss of numerical efficiency due to the dimensionality. A more significant problem is related to the imputation algorithms considered, whose complexity may be non-linearly dependent on the number of dimensions. In this case, dimp provides very good (i.e., linear) scaling.

However, future studies may compare more sophisticated methods using the benchmarks introduced. The same applies to the mechanisms causing the missingness. Instead of the analysed MCAR, other patterns, such as MAR and MNAR, can also be employed to check the imputation quality of fuzzy data.

Further steps concerning imputation procedures for fuzzy data are still necessary. New imputation procedures can still be designed, e.g., aimed at other kinds of fuzzy numbers, not just TRFNs and TPFNs. The same applies to benchmarking measures, e.g., when other distances and characteristics of FNs can be considered. Another idea is related to mixed types of variables in the same dataset, e.g., when some variables are real-valued while others are fuzzy-valued.

References

Afkanpour, M., Hosseinzadeh, E. and Tabesh, H. (2024). Identify the most appropriate imputation method for handling missing values in clinical structured datasets: A systematic review, *BMC Medical Research Methodology* **24**(1): 188, DOI: 10.1186/s12874-024-02310-6.

Amirfakhrian, M., Yeganehmanesh, S. and Grzegorzewski, P. (2018). A new distance on fuzzy semi-numbers, *Soft Computing* **22**(14): 4511–4524.

Asunción, A. and Newman, D. (2007). *UCI Machine Learning Repository*, `https://archive.ics.uci.edu/`.

Ban, A., Coroianu, L. and Grzegorzewski, P. (2015). *Fuzzy Numbers: Approximations, Ranking and Applications*, Polish Academy of Sciences, Warsaw.

Beck, M.W., Bokde, N., Asencio-Cortés, G. and Kulat, K. (2018). R package imputeTestbench to compare imputation methods for univariate time series, *The R Journal* **10**(1): 218–233.

Calcagnì, A. (2024). Estimating latent linear correlations from fuzzy frequency tables, *Communications in Mathematics and Statistics* **12**(3): 435–461.

Calcagnì, A., Grzegorzewski, P. and Romaniuk, M. (2025). Bayesianize fuzziness in the statistical analysis of fuzzy

data, *International Journal of Approximate Reasoning* **186**: 109495.

Colubi, A. (2009). Statistical inference about the means of fuzzy random variables: Applications to the analysis of fuzzy- and real-valued data, *Fuzzy Sets and Systems* **160**(3): 344–356.

Donders, A.R.T., van der Heijden, G.J., Stijnen, T. and Moons, K.G. (2006). Review: A gentle introduction to imputation of missing values, *Journal of Clinical Epidemiology* **59**(10): 1087–1091.

Dzulkalnine, M. and Sallehuddin, R. (2019). Missing data imputation with fuzzy feature selection for diabetes dataset, *SN Applied Sciences* **1**: 362.

Faraz, A. and Shapiro, A.F. (2010). An application of fuzzy random variables to control charts, *Fuzzy Sets and Systems* **161**(20): 2684–2694.

Fisher, R.A. (1936). The use of multiple measurements in taxonomic problems, *Annals of Eugenics* **7**(2): 179–188.

Gendre, M., Hauffe, T., Pimiento, C. and Silvestro, D. (2024). Benchmarking imputation methods for categorical biological data, *Methods in Ecology and Evolution* **15**(9): 1624–1638.

Gong, Y., Yang, S., Ma, H. and Ge, M. (2018). Fuzzy regression model based on geometric coordinate points distance and application to performance evaluation, *Journal of Intelligent and Fuzzy Systems* **34**(1): 395–404.

Grzegorzewski, P. (1998). Metrics and orders in space of fuzzy numbers, *Fuzzy Sets and Systems* **97**(1): 83–94.

Grzegorzewski, P., Hryniewicz, O. and Romaniuk, M. (2020). Flexible bootstrap for fuzzy data based on the canonical representation, *International Journal of Computational Intelligence Systems* **13**(1): 1650–1662.

Grzegorzewski, P. and Romaniuk, M. (2022). Bootstrap methods for epistemic fuzzy data, *International Journal of Applied Mathematics and Computer Science* **32**(2): 285–297, DOI: 10.34768/amcs-2022-0021.

Grzegorzewski, P. and Romaniuk, M. (2024). Bootstrapped tests for epistemic fuzzy data, *International Journal of Applied Mathematics and Computer Science* **34**(2): 277–289, DOI: 10.61822/amcs-2024-0020.

Hesamian, G., Akbari, M.G. and Shams, M. (2023). A goodness-of-fit test based on fuzzy random variables, *Fuzzy Information and Engineering* **15**(1): 55–68.

Jadhav, A., Pramod, D. and Ramanathan, K. (2019). Comparison of performance of data imputation methods for numeric dataset, *Applied Artificial Intelligence* **33**(10): 913–933.

Jäger, S., Allhorn, A. and Biessmann, F. (2021). A benchmark for data imputation methods, *Frontiers in Big Data* **4**: 693674, DOI: 10.3389/fdata.2021.693674.

Jordon, J., Szpruch, L., Houssiau, F., Bottarelli, M., Cherubin, G., Maple, C., Cohen, S.N. and Weller, A. (2022). Synthetic data—What, why and how?, `https://arxiv.org/abs/2205.03257`.

Kayacan, E. and Khanesar, M.A. (2016). Fundamentals of type-1 fuzzy logic theory, *in* E. Kayacan and M.A. Khanesar (Eds), *Fuzzy Neural Networks for Real Time Control Applications*, Butterworth-Heinemann, Oxford/Waltham, pp. 13–24.

Kim, B. and Bishu, R.R. (1998). Evaluation of fuzzy linear regression models by comparing membership functions, *Fuzzy Sets and Systems* **100**(1–3): 343–352.

Kowarik, A. and Templ, M. (2016). Imputation with the R package VIM, *Journal of Statistical Software* **74**(7): 1–16.

Kruse, R. (1982). The strong law of large numbers for fuzzy random variables, *Information Sciences* **28**(3): 233–241.

Kwakernaak, H. (1978). Fuzzy random variables. Part I: Definitions and theorems, *Information Sciences* **15**(1): 1–15.

Lee, K., Lim, H., Hwang, J. and Lee, D. (2024). Evaluating missing data handling methods for developing building energy benchmarking models, *Energy* **308**: 132979, DOI: 10.1016/j.energy.2024.132979.

Little, R. and Rubin, D. (2022). *Statistical Analysis with Missing Data*, John Wiley & Sons, Inc., New York.

Lubiano, M.A., Montenegro, M., Sinova, B., de la Rosa de Sáa, S. and Gil, M.A. (2016). Hypothesis testing for means in connection with fuzzy rating scale-based data: Algorithms and applications, *European Journal of Operational Research* **251**(3): 918–929.

Okuda, T., Kodono, Y. and Asai, K. (1990). Statistical estimation by fuzzy observation data, *Transactions of the Society of Instrument and Control Engineers* **26**(5): 564–571.

Parchami, A., Grzegorzewski, P. and Romaniuk, M. (2024). Statistical simulations with LR random fuzzy numbers, *Statistical Papers* **65**(6): 3583–3600.

Parchami, A., Grzegorzewski, P. and Romaniuk, M. (2025). Calculating probabilities with LR fuzzy random variables, *Soft Computing* **29**(15–16): 5129–5141.

Puri, M.L. and Ralescu, D.A. (1986). Fuzzy random variables, *Journal of Mathematical Analysis and Applications* **114**(2): 409–422.

Ramos-Guajardo, A., Blanco-Fernández, A. and González-Rodríguez, G. (2019). Applying statistical methods with imprecise data to quality control in cheese manufacturing, *in* P. Grzegorzewski *et al.* (Eds), *Soft Modeling in Industrial Manufacturing*, Springer, Cham, pp. 127–147.

Romaniuk, M. (2025a). Benchmarking imputation methods for fuzzy datasets—Supplementary material, `https://www.researchgate.net/publication/394962073_Benchmarking_Imputation_Methods_for_Fuzzy_Datasets_supplementary_material`.

Romaniuk, M. (2025b). *FuzzyImputationTest: Imputation Procedures and Quality Tests for Fuzzy Data*, `https://CRAN.R-project.org/package=FuzzyImputationTest`.

Romaniuk, M. and Grzegorzewski, P. (2023). Resampling fuzzy numbers with statistical applications: FuzzyResampling package, *The R Journal* **15**(1): 271–283.

Romaniuk, M. and Grzegorzewski, P. (2026). Fuzzy data imputation with dimp and FGAIN, *Journal of Computational Science* **93**: 102738.

Romaniuk, M., Grzegorzewski, P. and Parchami, A. (2024). FuzzySimRes: Epistemic bootstrap—An efficient tool for statistical inference based on imprecise data, *R Journal* **16**(2): 175–191.

Romaniuk, M. and Hryniewicz, O. (2019). Interval-based, nonparametric approach for resampling of fuzzy numbers, *Soft Computing* **23**: 5883–5903.

Romaniuk, M. and Hryniewicz, O. (2021). Discrete and smoothed resampling methods for interval-valued fuzzy numbers, *IEEE Transactions on Fuzzy Systems* **29**(3): 599–611.

Royston, P. (2004). Multiple imputation of missing values, *The Stata Journal* **4**(3): 227–241.

Schafer, J.L. (1997). *Analysis of Incomplete Multivariate Data*, Chapman and Hall/CRC.

Sefidian, A.M. and Daneshpour, N. (2019). Missing value imputation using a novel grey based fuzzy c-means, mutual information based feature selection, and regression model, *Expert Systems with Applications* **115**: 68–94.

Sinova, B., Vega, S.P. and Ángeles Gil, M. (2025). Strong consistency and robustness of fuzzy medoids, *International Journal of Approximate Reasoning* **182**: 109425.

Stekhoven, D.J. (2022). *missForest: Nonparametric Missing Value Imputation Using Random Forest*, `https://cran.r-project.org/package=missForest`.

Stekhoven, D.J. and Buehlmann, P. (2012). MissForest—Non-parametric missing value imputation for mixed-type data, *Bioinformatics* **28**(1): 112–118.

Tamaki, F., Kanagawa, A. and Ohta, H. (1998). Identification of membership functions based on fuzzy observation data, *Fuzzy Sets and Systems* **93**(3): 311–318.

Templ, M., Kowarik, A., Alfons, A., de Cillia, G., Prantner, B. and Rannetbauer, W. (2021). *VIM: Visualization and Imputation of Missing Values*, `https://cran.r-project.org/package=VIM`.

van Buuren, S. and Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in R, *Journal of Statistical Software* **45**(3): 1–67.

Williams, G.J. (2011). *Data Mining with Rattle and R: The Art of Excavating Data for Knowledge Discovery*, Use R!, Springer, New York.

Wilson, S. (2021). *miceRanger: Multiple Imputation by Chained Equations with Random Forests*, `https://CRAN.R-project.org/package=miceRanger`.

Yeganehmanesh, S., Amirfakhrian, M. and Grzegorzewski, P. (2018). Fuzzy semi-numbers and a distance on them with a case study in medicine, *Mathematical Sciences* **12**(1): 41–52.

Zhang, S. (2012). Nearest neighbor selection for iterative kNN imputation, *Journal of Systems and Software* **85**(11): 2541–2552.

Maciej Romaniuk received his MSc degree in mathematics from the Faculty of Mathematics, Informatics, and Mechanics, University of Warsaw, Poland, in 2001. He earned his PhD and DSc (habilitation) degrees in computer science from the Systems Research Institute, Polish Academy of Sciences, in 2007 and 2018, respectively. The main topics of his current research are Monte Carlo simulations, simulations of events under circumstances of uncertain and imprecise information, statistics, fuzzy numbers as well as financial and actuarial mathematics.

Received: 27 August 2025
Revised: 8 January 2026
Re-revised: 16 February 2026
Accepted: 17 February 2026