Statistics and Computing (2024) 34:121
https://doi.org/10.1007/s11222-024-10429-1

ORIGINAL PAPER

Robust multipe imputation with GAM

Matthias Templ[1]

Received: 8 January 2024 / Accepted: 27 March 2024 / Published online: 22 May 2024

Abstract
Multiple imputation of missing values is a key step in data analytics and a standard process in data science. Nonlinear imputation methods come into play whenever the linear relationship between a response and predictors cannot be linearized by transformations of variables, adding interactions, or using, e.g., quadratic terms. Generalized additive models (GAM) and its extension, GAMLSS—where each parameter of the distribution, such as mean, variance, skewness, and kurtosis, can be represented as a function of predictors, are widely used nonlinear methods. However, non-robust methods such as standard GAM's and GAMLSS's can be swayed by outliers, leading to outlier-driven imputations. This can apply concerning both representative outliers—those true yet unusual values of your population—and non-representative outliers, which are mere measurement errors. Robust (imputation) methods effectively manage outliers and exhibit resistance to their influence, providing a more reliable approach to dealing with missing data. The innovative solution of the proposed new imputation algorithm tackles three major challenges related to robustness. (1) A robust bootstrap method is employed to handle model uncertainty during the imputation of a random sample. (2) The approach incorporates robust fitting techniques to enhance accuracy. (3) It effectively considers imputation uncertainty in a resilient manner. Furthermore, any complex model for any variable with missingness can be considered and run through the algorithm. For the real-world data sets used and the simulation study conducted, the novel algorithm *imputeRobust* which includes robust methods for imputation with GAM's demonstrates superior performance compared to existing imputation methods using GAMLSS. Limitations pertain to the imputation of categorical variables using robust techniques.

Keywords Missing values · Multiple imputation · Generalized additive models · Robust estimation · Robust bootstrap

1 Introduction

The imputation of missing values is undoubtedly an essential part of data science applications. It is listed in data mining project model frameworks such as CRISP (Chapman et al. 2000) or GSBPM (Vale 2009) with data cleaning being one of the essential phases.

The current literature on data cleaning also reports that missing values are crucial to address (Rahm and Do 2000). It shows the imputation of missing values as one integrative part (Mavrogiorgou et al. 2021) and addresses the problems of missing values; see, e.g., Brownlee (2020). In addition, other researchers also explore the use of robust imputation techniques (Templ et al. 2011; Templ 2023) in order to mitigate the impact of outliers during the imputation process, however, it should be noted that these methods are limited to linear relationships.

1.1 Types of missing data

Various types of missing data exist, but for the scope of this study, we focus solely on one type. To elucidate the distinctions, we highlight both MAR (Missing At Random) and MCAR (Missing Completely At Random).

Data is MCAR when missingness is equally likely across all units, i.e. if

$$P(R = 1|Y_{obs}, Y_{mis}, \boldsymbol{\phi}) = P(R = 1, \boldsymbol{\phi})$$

This means that the missingness event, R (with $R = 1$ for observed and $R = 0$ for missing), does not depend on both seen (Y_{obs}) and unseen (Y_{mis}) data. The parameter $\boldsymbol{\phi}$ represents unidentified factors that affect the data or its missing values.

✉ Matthias Templ
matthias.templ@fhnw.ch

[1] Institute for Competitiveness and Communication, University of Applied Sciences and Arts Northwestern Switzerland, Riggenbachstrasse 16, 4600 Olten, Switzerland

Data is MAR if missingness depends on observed data:

$$P(R = 1|Y_{obs}, Y_{mis}, \boldsymbol{\phi}) = P(R = 1|Y_{obs}, \boldsymbol{\phi})$$

1.2 Imputation uncertainty, multiple imputation and model uncertainty

In imputing missing data, we face intrinsic uncertainties, commonly grouped into two categories: imputation uncertainty and model uncertainty, as highlighted by (van Buuren 2012; Templ 2023). Moreover, multiple imputation does not provide a fixed value but generates several estimates from a distribution for each missing entry.

Imputation uncertainty Uncertainty in imputed values arises from our guesses about missing values, based on observed data and our assumptions. These guesses are not definite, as real values could differ. Rather than replacing the missing data with a single prediction (e.g., the mean), we introduce noise or draw from a predictive distribution. For example, in addressing missing values via linear regression, we might add a random residual to the predicted mean, termed stochastic regression imputation, capturing this uncertainty.

Multiple imputation Considering uncertainties, we impute with noise rather than fixed values, acknowledging that missing data should be treated as a distribution, not a constant. Multiple imputation addresses this by generating several datasets, each replacing missing values with plausible entries based on observed data. Analyzing these datasets and pooling the results offers an outcome that considers potential variability in the missing data.

Model uncertainty Model uncertainty is relevant for random samples but not for population/census data. It relates to the assumptions made when using a model, like linear regression, to predict missing values. These assumptions might not reflect the true data process. Bayesian methods address this by modeling the posterior distributions of missing data, explicitly incorporating model uncertainty through priors and posterior distributions of parameters. While bootstrapping, introduced in the context of multiple imputation in a series of papers for parametric models (see, e.g., Honaker and King 2010) and resulting in the R package Amelia II (Honaker et al. 2011), does not directly compute posterior distributions, it approximates variability in estimates to reflect sampling uncertainty. The choice between Bayesian methods and bootstrapping to handle model uncertainty depends on the specifics of the analysis, the nature of the data, the method used and the preference of the analyst. However, Bayesian regression and posterior parameter draws are limited in complex methods such as GAM.

It should be noted that in the literature, bootstrapping was utilized either before or after imputation. We used the bootstrap before imputation as proposed by Shao and Sitter (1996) and Little and Rubin (2002). When the imputation and analysis procedures are uncongenial and/or misspecified (Meng 1994), only the bootstrap before multiple imputation approach and the method of von Hippel (von Hippel and Bartlett 2021) give intervals with nominal coverage (Bartlett and Hughes 2020).

By considering both imputation and model uncertainties (through bootstrapping), we achieve more realistic inferences. Imputation methods, especially in multiple imputations, should incorporate randomness to address both uncertainties.

1.3 The problem with outliers

Statistical analyses can be significantly influenced by outliers. They can be categorized into two types: representative and non-representative outliers (Chambers 1986).

Representative outliers are genuine observations distant from the bulk of the data, like billionaire incomes in household data. They are genuine and removing them may bias results by reducing observed variability. Yet, they can dramatically sway non-robust imputation methods. For instance, they might distort a regression fit, leading to unrealistic imputed values.

Non-representative outliers stem from errors like data entry mistakes or anomalies, like misrecording a height as 250 cm instead of 150 cm. They should be addressed in non-robust imputation methods to prevent misleading data patterns. Detecting and replacing them is challenging (Filzmoser and Gregorich 2020), so initial use of robust imputation methods is preferable.

Differentiating between these outliers can be tough, and removing them is not advised due to reduced sample size impacting variance estimation (Templ et al. 2019). Utilizing robust imputation methods bypasses the need for outlier detection and differentiation. Both outlier types can skew non-robust methods, necessitating down-weighting to mitigate their influence. Without this, outliers might entirely dictate fits, leading to unrealistic imputed values.

Excluding outliers for a better non-robust fit isn't ideal as it reduces sample size and identifying these outliers is tough, and, practically, it may be difficult to judge what is an outlier and what is not an outlier. However, robust statistical methods help with this issue and downweight outliers that have a large influence on the estimation.

Note that inconsistencies in the data, such as a 6-year-old girl classified as married, are not considered in the robust GAM framework, and we refer to error location and deductive imputation methods (van der Loo and de Jonge 2018).

1.4 Outline

Initially, GAMs are briefly presented in Sect. 2.1, which is followed by an exploration of established methods, partic-

ularly GAMLSS for handling missing values, as defined in Sect. 2.2. This segment already offers a visual representation that underscores the need for a new innovative approach to impute missing values using GAM, given the excessive scatter observed in multiple imputations. The novel strategy and methodology are detailed in Sect. 2.3. This includes the use of GAMs combined with *thin plate regression splines* within a Bacon algorithm for enhanced robustness. Furthermore, a robust bootstrap is adopted to accommodate model uncertainties, while imputation uncertainties are tackled using a variety of techniques including *predictive mean matching* (PMM) and *midastouch*. Standard evaluation metrics are used; see Sect. 3, to judge the quality of imputations. These include precision measures, but also confidence interval coverage rates of an estimator. Section 4 shows the results on real and simulated data, and Sect. 5 concludes.

2 Imputation with GAM

Multiple Imputation by Chained Equations (MICE), also known as Fully Conditional Specification (FCS), is a popular framework for handling missing data. This method works by performing multiple imputations for the missing values (by default, it uses bootstrap samples from the data), creating several different complete data sets. The results of these data sets can then be pooled to create a single estimate.

PMM finds for each missing value, find a set of observed values with closest predicted means and randomly draw an observed value from this set to impute the missing value. The method *midastouch*, on the other hand, also fits a linear regression model to predict missing values but it modifies the selection criteria by converting the distances between each predicted y_{obs} and the corresponding predicted y_{mis} to probabilities. Whilst PMM considers only neighbors, in midastouch a donor is drawn from the entire donor pool considering the drawing probabilities.

2.1 GAM

Non-linear methods such as GAM are important kind of methods in statistical modeling, and for many data sets GAM can be the preferable method to model a response with possible non-linear relationships to predictors. So, as they are useful in practice to describe a response, they are as useful for the imputation of missing values.

GAM represents an extension of the generalized linear models (GLM) that allow for more flexibility in modeling the relationship between the response and predictor variables, especially for non-linear relations between expected values and their functional relationship to the predictors. In other words, the linear assumption can be overly restrictive.

GAMs address this limitation by incorporating non-parametric smooth functions for one or more predictors, thereby accommodating non-linear relationships. Instead of modeling the response as a linear combination of predictors, GAM uses a sum of smooth functions of predictors. Symbolically, a GAM might be expressed as

$$E(Y) = \beta_0 + s_1(X_1) + s_2(X_2) + \cdots,$$

where $E(Y)$ is the expected value of the response variable Y, and $X_1, X_2, \ldots$, are predictor variables, β_0 represents an intercept, and $s_1, s_2, \ldots$, are smoothing functions.
The primary advantages of GAMs are:

1. *Flexibility*: They can capture a wide range of non-linear relationships without having to specify the form of the relationship a priori.
2. *Interpretability*: Although more flexible than GLMs, GAMs retain a level of interpretability because they model the relationship between the response and each predictor separately, allowing for clear visualization and interpretation of individual effects.
3. *Generality*: They can handle different types of response distribution (e.g. normal, binomial, Poisson) in a manner analogous to GLMs.

GAMs offer a powerful and flexible modeling approach that bridges the gap between the simplicity of linear models and the flexibility of fully non-parametric models. They are particularly useful in scenarios where the relationship between predictors and the response is believed to be nonlinear, but where the exact form of this nonlinearity is unknown.

The use of thin plate regression splines as smoother was first proposed by Wood (2003). For a detailed explanation of this method, we recommend referring to the original source, as it is beyond the scope of this paper to formally introduce its intricacies. Using thin plate regression splines has several advantages and involves additional considerations. Computational costs are reduced by using a low-rank approximation to thin plate regression splines, which provides a balance between flexibility and computational efficiency. Additionally, a penalty as regularization step is applied to the "wiggliness" of the spline to prevent overfitting. The strength of the penalty is determined by the data and is estimated as part of the model-fitting process using restricted maximum likelihood. Thin plate regression splines can also be used for bivariate or even trivariate smoothing. Thin plate regression splines have isotropic properties. Moreover, the reliable automatic choice of the number and position of the knots, therefore, the automatic choice of smoothing parameters, is one of the main advantages of the implementation of Wood (2003) in Wood (2006) and the reason why it is used within our new imputation method.

2.2 GAMLSS for missing values

The GAMLSS method (Rigby and Stasinopoulos 2005; Stasinopoulos et al. 2017) extends the generalized additive model. A unique feature of GAMLSS is its ability to specify a (nonlinear) model for each of the parameters of the distribution, thus giving rise to an extremely flexible toolbox that can be used to model almost any distribution.

Roel de Jong and Spiess (2016) developed a series of GAMLSS-based imputation methods, so it is now easy to perform multiple imputation under a variety of distributions. The ImputeRobust package (Salfran and Spiess 2018), implements various methods for continuous data that can be directly used with the multiple imputation framework of R package mice (van Buuren 2012), e.g., `gamlss` (normal distribution), `gamlssJSU` (Johnson's SU distribution), `gamlssTF` (t-distribution) and `gamlssGA` (Gamma distribution).

The imputation algorithm generates a random function for missing values by drawing a bootstrap sample from the fully observed data's GAMLSS model, all within a multiple imputation by chained equations context. It is important to highlight that when using the GAMLSS method for imputation in the ImputeRobust R package (Salfran and Spiess 2018), there is a parameter to adjust for extreme values, deciding whether to rectify extreme imputed values. Although we consistently set this parameter to correct for any extreme imputed values, often some extreme values remained.

In Fig. 1 we use the ethanol data set that includes a few outliers (in green) and two data points were set as missing in NOx, because the implementation of the GAMLSS imputation procedure reports errors when only one value is set to missing. Anyhow, it gives a better picture on two well-chosen points to be set to missing—one on the peak of the curve, and one on the non-linear decreasing part. We will now examine the various methods used to impute the two missing values in NOx and visualize the distribution of these multiple imputed values. Consequently, we have one distribution curve representing the imputed values corresponding to the peak of NOx values, and another curve representing the imputed values for the second data point that was set to missing. In addition, all multiple imputed values are shown for both missing values in NOx.

Clearly, Fig. 1 already shows the problems of imputation with imputation using GAMLSS. The distribution of multiple imputed values is far too large. There are, in fact, two problems. First, a few extreme values are imputed even with the setting of correcting them within the procedure. Secondly, even after deleting those very extreme imputed values, the distribution of imputed values is (still) far too large; see the third and fourth graphics in Fig. 1. This figure clearly demonstrates that the imputations provided by GAMLSS are not useful. Note that the remaining graphics are discussed later.

2.3 A new proposal to impute with GAM

Bacon, short for Blocked Adaptive Computationally-Efficient Outlier Nominators (Billor et al. 2000), is a somewhat robust algorithm. The Bacon algorithm proved to be one of the most robust and fastest forward search method.

The algorithm starts from an initially small subset of non-outlier ("good") data and repeatedly adds to the subset those observations whose distances are smaller than a predefined threshold (see also Schoch 2021). The observations that are not included in the final subset are referred to as outliers. The algorithm terminates if the subset cannot be increased any further. The original Bacon algorithm for multivariate outlier nomination can be initialized in two ways: Version "V1" or "V2" (Billor et al. 2000). In version V2, the algorithm starts with the coordinate-wise median. The breakthrough point, as indicated by Bacon (2000), is approximately 40%. However, it is important to note that these estimates do not possess affine-equivariant properties in terms of population location and dispersion. Version V1 exhibits affine-equivariant characteristics, which is often a desired condition in multivariate statistics. But, since it starts from the coordinate-wise median, the estimates have a very low theoretical breakthrough point (Schoch 2021), a measure that indicates the smallest proportion of contamination (erroneous or extreme values) that can cause an estimator to give arbitrarily large incorrect results. However, in particular, the breakthrough point proved to be high in practical applications (Todorov et al. 2011; Grentzelos et al. 2021), and realistically it does not converge to a "robust" solution if the number of outliers is both very high and very far from the main bulk of the data.

This general procedure has been adapted in Algorithm 1 to robustify GAM. This is more or less related version V1, as it starts non-robustly.

In comparison to other Mahalanobis methods, Bacon also exhibits excellent performance (Cédric and Beat 2008). The Bacon method is particularly suitable for large data sets. The Bacon algorithm was competitive and often showed the best performance in other studies Billor et al. (2000).

The imputation algorithm consists of several steps. The first step represents the basic chain and represents the well-known MICE iterative EM-algorithm and is roughly outlined in the Algorithm 2.

The process alternates between estimating the model and imputing data until the variations in the imputed values fall below a designated threshold, signaling that convergence has been achieved. Multiple imputations can be performed by repeatedly calling the Algorithm 2.

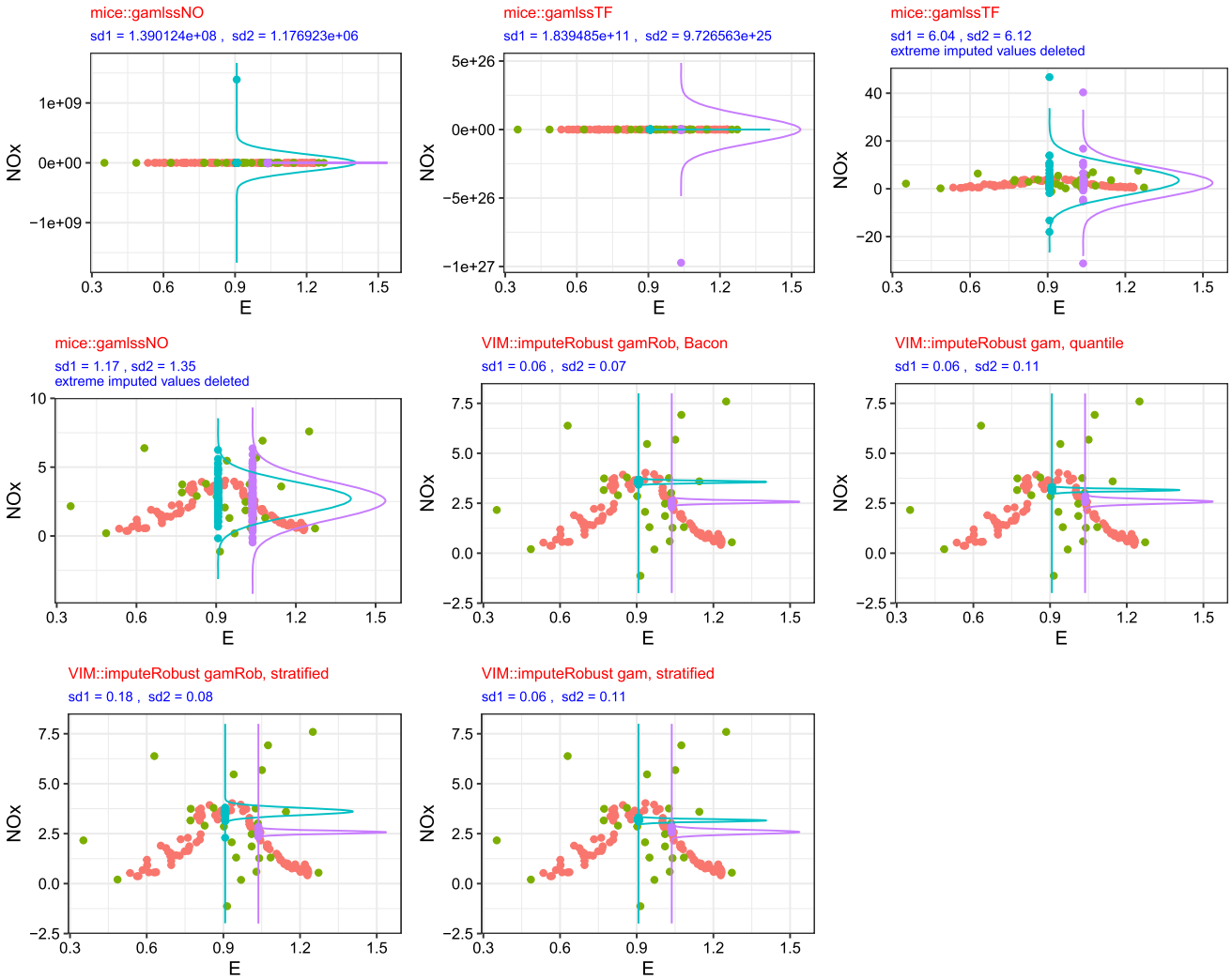


Fig. 1 Ethanol data set with few outliers (in green). Distribution of multiple imputations of two data points with GAMLSS (NO and TF) and from the new proposed imputation method (non-robust and robust versions) with different bootstrapping approaches (gam: quantile, gam: stratified, gamRob: Bacon, gamRob: stratified). For our proposed imputation method of GAM (gam) and robust GAM (robGAM), the PMM method was used to consider imputation uncertainty (colour figure online)

Considering model uncertainty can be achieved by drawing bootstrap samples from the data and applying the model to these samples.

However, even when robust statistical methods are utilized to fit a model, the bootstrap approach can encounter issues with datasets containing outliers. The presence of numerous outliers can lead to their overrepresentation in bootstrap samples. Specifically, since bootstrapping involves resampling with replacement, outliers might be chosen repeatedly in the resampled dataset. This repeated selection can skew the bootstrap distribution due to the excessive influence of these outliers.

The robust bootstrap technique, introduced by Salibián-Barrera et al. (2008), modifies the conventional bootstrap approach to mitigate the impact of outliers on the bootstrap distribution. This method begins by securing an estimate of the target parameter. Generally, the goal of reducing the effects of outliers can be reached with different procedures.

(a) Obtaining bootstrap samples by reducing the sampling weight of the outliers.
(b) Obtaining bootstrap samples by sampling with a weight of outliers equal to zero.
(c) Alternatively, observations can be stratified according to certain quantiles of their residuals. The bootstrap then selects bootstrap samples in each strata independently. This (also) allows outliers to be not over-represented in a bootstrap sample.

Algorithm 1 Robust GAM

Require: $formula, data, fraction = 0.5, max\_iter = 10$
function ROBGAM($formula, data, fraction, max\_iter$)
  $response\_name \leftarrow$ first variable of $formula$
  $predictor\_names \leftarrow$ all variables of $formula$ except the first
  $gam\_model \leftarrow$ fit GAM using $formula$ and $data$ ▷ Fit the initial GAM
  for $i \leftarrow 1$ to max_iter **do**
    $residuals \leftarrow$ compute residuals from gam_model using $data$ and $predictor\_names$
    $squared\_residuals \leftarrow residuals^2$ ▷ Calculate squared residuals
    $ranks \leftarrow$ rank of $squared\_residuals$
    $h \leftarrow \lfloor fraction \times$ number of rows in $data \rfloor$
    $subset \leftarrow$ data with ranks less than or equal to h
    $gam\_model\_new \leftarrow$ fit GAM using $formula$ and $subset$ ▷ Refit the model with the subset of data
    if all absolute differences between coefficients of gam_model and gam_model_new are less than $1e-5$ **then**
      break
    else
      $gam\_model \leftarrow gam\_model\_new$
    end if
  end for
  return gam_model_new
end function

Algorithm 2 Iterative Model-Based Imputation - basic chain

Take a (robust) boostrap sample of your data (details follow later)
Initialize: Impute missing values using kNN imputation.
repeat
  For each variable j with missing values do:
    (a) Model j using GAM or robust GAM from algorithm 1. A model can be specified for each variable.
    (b) Predict the missing values in j using the estimates from the model.
  Update: Replace the missing values in variable j with the noised predictions. This noise is adequately chosen to consider imputation uncertainty.
until convergence (changes in the imputed values fall below a specified threshold)

(d) Stratify in only two groups separated based on the quantile Q_{alpha}, with $alpha$ the assumed percentage of *good data points*.

In Algorithm 3 this approach is adapted to consider the uncertainty of the model. Both bootstrap methods help to ensure that the bootstrap distribution is more representative of the underlying distribution of the data, without being overly influenced by outliers or high-leverage points.

nameAlgorithm imputeRobust (bootstrap part) 3

imputeRobust is an implementation for multiple imputation of missing data. The code aims to estimate the missing values in a data set using robust techniques.

The input parameters to the *imputeRobust* algorithm are

formulas: This is a list comprising various models. For every variable present in the data, an intricate model formula can be specified. In the absence of a provided formula, the most basic model is employed, which explains the response using all the predictor variables found in the dataset.

Algorithm 3 imputeRobust (bootstrap part) 3 Robust Bootstrap for imputation with GAM using continuous weights.

procedure ROBUSTBOOTSTRAP($Data, NumBootstrapSamples$)
  Compute GAM on $Data$, obtaining residuals.
  for $i = 1$ to $NumBootstrapSamples$ **do**
    Generate a bootstrap sample from $Data$ with weights (probabilities to be sampled) based on the residuals. Outliers, defined by large residuals, should have smaller weights using the robustness weights $\phi(r_i/S)/(r_i/S)$ with r_i the residuals, S the robust scale estimate of residuals and ϕ the Tukey's biweight function (Beaton and Tukey 1974)).
    Compute GAM on the bootstrap sample.
  end for
end procedure

Algorithm 4 Robust Bootstrap for imputation with GAM using stratification.

procedure ROBUSTBOOTSTRAP($Data, NumBootstrapSamples$)
  Compute GAM on $Data$, obtaining residuals.
  Stratify observations in H strata using certain quantiles of the residuals. By default option (d) from aboves description is used with $\alpha = 0.75$.
  for $i = 1$ to $NumBootstrapSamples$ **do**
    Draw H bootstrap samples from $Data$ in each strata independently and join them together.
    Compute GAM on the joined bootstrap sample.
  end for
end procedure

data: This refers to the dataset that contains the missing values awaiting imputation.

boot: logical flag to use bootstrapping.

robustboot: kind of robust bootstrap (residual or stratified).

alpha: if robustboot is equal stratified, then alpha is the chosen ratio of the number of non-outlying/good data points in a data set.

method: This specifies the regression technique used to impute variables. The available methods include LTS regression, ordinary least squares estimation, GAM with thin plate splines, robust GAM with thin plate splines, and MM regression.

multinom.method: Refers to the approach for multinomial logistic regression, which is utilized when imputing nominal variables. For this, multinomial log-linear models facilitated by neural networks are used (Venables and Ripley 2002).

eps: A numerical threshold that defines the stopping criterion for the iteration process.

maxit: Represents the upper limit on the number of iterations the algorithm can perform.

uncert: A descriptor indicating the approach to introduce uncertainty into the imputed values. Options include the use of normal errors, residual errors, midastouch, or the PMM method. Refer to Algorithm 6 for a comprehensive understanding.

The primary procedures of the algorithm are detailed in the Algorithms 5 and 6.

nameAlgorithm

Algorithm 5 imputeRobust

procedure IMPUTEROBUST(formulas, data, boot, robustboot, method, multinom.method, takeAll, eps, maxit, alpha, uncert, familiy, value_back, trace)
  Set initial parameters and convert character variables into factors.
  Identify the type of factor (either dichotomous or polytomous) and recognize any problems with factors (e.g., factor levels without any observation)
  if takeAll is true **then**
    If the takeAll option is activated, initialize the missing values using kNN imputation, as described in Kowarik and Templ (2016); Templ (2023).
  end if
  while criteria > eps and iterations < maxit **do**
    For any variable j with missing values do:
    if if boot or robustboot is true **then**
      draw a (robust) bootstrap sample considering Algorithms 3 or 4. Fit the following model on the boostrap sample.
    end if
      (a) Apply (roubust) GAM using thin plate regression splines with response j on all other variables using observations where j is not missing.
      (b) Predict the missing values in j with the model obtained in (a).
    Update: Substitute the missing entries in variable j with predictions that have been adjusted with appropriate noise. This noise selection ensures that the imputation uncertainty is taken into account, whether by introducing random normal disturbances, selecting residuals, employing PMM, or using midastouch.
    Refresh the criteria value until a state of convergence is achieved, where the modifications in the imputed values are minimal and below a set threshold.
  end while
  return imputed data.
end procedure

nameAlgorithm imputeRobust (cont.)

Algorithm 6 Algorithm imputeRobust (cont.) Imputation uncertainty

procedure IMPUTATIONUNCERTAINTY(uncert)
  if response is continuous **then**
    if uncertainty corresponds to the normal error **then**
      It produces random numbers following a normal distribution with a mean of 0 and a (robust) standard deviation. These random numbers are then added to the predicted values, meaning that the predicted values are adjusted with normally distributed errors.
    end if
    if uncertainty corresponds to residual error **then**
      Draws residuals from the fitted model and adds them to the expected values. For robust imputation, only residuals from non-outlier data points are selected. This approach presumes that the model's residuals can encapsulate the variability of the missing values.
    end if
    if uncertainty corresponds to weighted residual error - midastouch **then**
      Assigns weights to residuals based on their proximity to the missing data point and selects a residual to add to the expected value. This sophisticated method assigns weights to residuals according to their relative distance in the predictor space from the imputation point. Residuals are chosen with these weights to determine their likelihood of selection.
    end if
    if uncertainty equals PMM **then**
      This refers to the widely recognized PMM technique. For every missing entry, the observed data points with the nearest predicted values are identified, and an observed value from this set is randomly chosen to serve as the imputed value.
    end if
  end if
  if response is nominal **then**
    Select a category by sampling, using the fit of the model to determine the probability that each category is chosen.
  end if
  return Imputed values.
end procedure

3 Judging the imputation algorithms

Both the precision and the uncertainty (and bias) of the estimators play complementary roles. Precision focuses on how close the imputed values are to the actual values, reflecting the method's ability to reproduce an existing data set accurately. The confidence intervals and mean squared errors of the estimators address the reliability and stability of the imputation method in terms of statistical inference and decision-making under uncertainty.

When evaluating imputation methods, considering both precision and confidence intervals ensures a balanced approach, acknowledging both the need for accuracy and the inherent uncertainty of dealing with incomplete data.

If the imputed data were completely different from the observed data, but the estimates were unbiased for all relevant analyses, we would be more or less satisfied. This argument is often used in the multi-imputation community to argue against precision measures in favor of only investigating the bias and variances of estimators (see, e.g., van Buuren 2012).

However, with complex data, it is possible that we do not know the relevant analyses to be performed by the users. The assumption of which estimators are relevant for the underlying data set is too strong since it is often not known which estimator is of interest. For example, official statistics data are often complex and often include hundreds of variables, and it is not known what researchers will analyze and which estimators they are interested in. It would be beyond the scope of any project to review dozens or even hundreds of estimators.

Furthermore, e.g., coverage rates are typically calculated in simulations (and simulated data) and not from real data.

Therefore, precision measures that show a general trend of whether imputations are reasonable are also helpful, especially within replications, when missing values are repeatedly placed on observed complete data. So, it is important to measure whether imputation methods introduce systematic bias on estimates, but it is also of interest to show if imputed values merely provide the same data structure as the original data within a simulation setting, especially because one hardly can show unbiasedness of all relevant estimators since data producers are not able to know which estimators users want to apply and thus a selection of a few ones cannot give a clear picture (Templ 2023).

3.1 Precision measures

Given a variable $y = (y_1, y_2, ..., y_n)$ with true values and $\hat{y} = (\hat{y}_1, \hat{y}_2, ..., \hat{y}_n)$ as the corresponding imputed values, the MAPE for a missing variable is calculated as

For a variable $y = (y_1, y_2, ..., y_n)$ with actual observed true values and $\hat{y} = (\hat{y}_1, \hat{y}_2, ..., \hat{y}_n)$ representing the imputed values, the Normalized Root Mean Squared Error (NRMSE) as a standardized measure of the differences between the actual values and the imputed values from a model (Stekhoven and Bühlmann 2011) is calculated with

$$NRMSE = \sqrt{\frac{1}{n_{(miss)}} \sum_{i=1}^{n_{(miss)}} \left(\frac{y_i - \hat{y}_i}{s} \right)^2} \quad , \tag{1}$$

where $n_{(miss)}$ is the total number of missing values and where s is the standard deviation of the true values. If there is more than one variable with missings, the NRMSE is calculated for each variable and summed up.

The Mean Squared Error of Correlation (MSECor) quantifies the average squared deviation between the correlations present in two distinct data sets or two variations of the same data set. This metric is often used to assess the efficacy of an imputation technique in retaining the inherent correlation structure of the primary data. The computation for MSECor is as follows (Templ et al. 2011):

$$MSECor = \frac{1}{p} \sum_{i=1}^{p} \sum_{j=1}^{p} (cor(X_i, X_j) - cor(Y_i, Y_j))^2 \quad , \tag{2}$$

where p is the number of variables in the data set. The coefficient $cor(X_i, X_j)$ represents the correlation between the i-th and j-th variables in data set X, while $cor(Y_i, Y_j)$ denotes the correlation between the i-th and j-th variables in data set Y (which is the data set with imputed values).

3.2 Uncertainty and bias of estimators

The coverage rate for an estimator is determined by examining whether the true parameter value, which is only known during a simulation, falls within a $(1-\alpha)\%$ confidence interval surrounding the estimated parameter. This is done by continually simulating data, imputing missing values, and computing the estimator and its confidence interval. The well-known coverage rate is expressed as

$$CR = \frac{1}{R} \sum_{i=1}^{R} I(l_i \leq \theta \leq u_i) \quad , \tag{3}$$

where R denotes the total number of simulations, I is an indicator function that takes the value 1 when the condition $l_i \leq \theta \leq u_i$ holds true and 0 otherwise. Here, l_i and u_i are the confidence interval's lower and upper limits, respectively, for the estimator θ.

It is worth noting that this definition incorporates non-representative outliers. Therefore, after imputation, the coverage rate and its corresponding confidence intervals are computed using only the observations that are not non-representative outliers.

The root mean squared error (RMSE) of an estimator, assessed by simulation, is a popular metric to gauge the discrepancies between the values forecasted by a model or estimator and the actual observed values. Within a simulation framework, the simulation might be executed multiple times using varied random inputs. During each execution, the difference between the estimator's prediction and the true value is observed. This difference, or error, is squared for every simulation iteration. The average of these squared errors over all iterations is then computed. Taking the square root of this mean provides the RMSE. Thus, given that $\hat{\theta}$ symbolizes the estimator and θ denotes the genuine value, the RMSE over n simulation iterations is expressed as:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{\theta}_i - \theta)^2} \tag{4}$$

This equation offers insight into the precision of the estimator. It indicates the typical estimator's deviation after imputation from the true values. The lower the RMSE the better.

3.3 Data sets and simulation settings

The following data sets are used to compare the methods:

Ethanol: Engine exhaust fumes from burning of ethanol. Ethanol fuel was burned in a single-cylinder engine. For

various settings of the engine compression and equivalence ratio, the emissions of nitrogen oxides were recorded. This data set with 88 observations was published in Brinkman (1981) and up to 25 moderate outliers have been added (see Figs. 1 and 2), see the graphics in the upper right of Fig. 2. The data set size is 113 observations and two variables (NOx and E).

Dolphin: The Galveston Ship Channel (GSC) is a bustling waterway frequented by both human activities, including shipping, fishing, and dolphin tourism, and by common bottlenose dolphins. A study conducted between June and August 2013 used a digital theodolite to track and observe dolphin behaviors in relation to vessel movements. In particular, dolphin behavior exhibited significant variations based on factors such as time of day, group size, and the presence of boats and calves and their swimming speeds and their direction change rate (reorientation rate) increased in the vicinity of tour boats and trawlers. Such changes in behavior could pose immediate and long-term risks to dolphin health and survival. The data consist of 167 observations and 11 variables such as swimming speed, reorientation rate, linearity (of swimming), distance, time, group size, calf presence, behavior, type of boat and number of boats. GAM was applied in Piwetz (2019). In each simulation, 20% missing values were randomly inserted in variable 'speed' with MAR related to variable 'reorientation rate' (sample probability weights linear to values in the reorientation rate).

Dolphin with outliers: This is the same data set and missing values inclusion, but 40 outlier data points will be generated and appended. A 5x5 diagonal covariance matrix is created for variables speed, rr, lin, distance and timeper. The upper and lower triangular elements of this matrix are then set to 0.9, creating a specific covariance structure while the diagonal elements are 1. This covariance matrix together with the mean vector of $\mu = (6, 70, 1, 300, 1)$ is used to generate 40 data points from a multivariate normal distribution. Random normal noise with mean 0 and variance $\sqrt{30}$ is added to the reorientation rate and distance columns of those 40 observations. The 40 values of the variables, type of boat, group size, calf presence, behavior, and number of boats are just sampled with replacement variable by variable from the original data. This results in moderate outliers near the main data cloud.

Simulated (non-linear) data I: This situation pertains to the utilization of non-linear data. The model for data generation varies from the model used for analysis in this particular scenario. Data sets with 200 observations and five variables are simulated using the approach of Gu and Wahba (1991) implemented in the R package mgcv (Wood 2006) in their function 'gamSim'. 10% missing values are inserted into the first variable using MCAR. For scenarios with outliers, 25 outliers are added to the previous data set by using the correlation of the 200 non-outliers plus a shifted mean vector by adding $(-0.35, -0.01, 0.5, -0.5, 0.5)$. In addition, random noise is added to the second column of the outlier data. This noise comes from a normal distribution with mean 0 and standard deviation 2. Moreover, for the recently added outlier, the values in the first and forth column are modified. This is done by adding values sampled from a bivariate normal distribution with the specified mean $(-8, 3)$ and covariance structure with 0.8 correlation. With this design, we produce moderate outliers, thus not far from the main bulk of the data, in the response variable and in the predictor space.

Simulated data II The situation involves a basic data simulation using a multivariate normal distribution, including missing data (MAR) and the presence or absence of outliers. The objective is to assess if the coverage rates align with the specified significance level (0.05 or 95% confidence intervals). Let

$$\sigma = \begin{pmatrix} 1 & r & 0 \\ r & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \mu = \begin{pmatrix} 5 \\ 10 \\ 10 \end{pmatrix}, \mu_{out} = \begin{pmatrix} 5+3.2 \\ 10+0.1 \\ 10-0.25 \end{pmatrix}$$

where $r \in \{0.5, 0.95\}$.
The regular data points are now drawn with $X_{reg} \sim \mathcal{N}(\mu, \sigma)$ while the outliers are drawn with $X_{out} \sim \mathcal{N}(\mu_{out}, \sigma/2)$
This gives outliers not far from the *cloud* of the non-outliers, thus the choices of values are based on explorative comparisons between the outliers and non-outliers so that both groups are not far from each other but to guarantee seperation between them.

Simulated data III: The data generation is designed to simulate a data set for linear regression analysis. The analysis and data generation model remained unchanged.

1. The number of predictor variables is set to $p = 4$.
2. Set the target R^2 value to a predefined value. We use $R^2 = 0.5$ and $R^2 = 0.95$.
3. Allocate a portion of R^2 to the first predictor x_1, denoted as $R^2_{x1} = R^2 \times 0.8$.
4. Define the regression coefficients as $\beta = \left[1, \frac{\sqrt{2}}{3}, \frac{\sqrt{2}}{3}, \frac{\sqrt{2}}{3}\right]$.
5. Generate the matrix of predictor variables X from a standard multivariate normal distribution of dimension $n \times p$.
6. Standardize X to have unit variance for each predictor.
7. Adjust the variance of x_1 to match R^2_{x1}.
8. Generate the error term, ϵ, with a standard deviation to ensure the desired R^2 for the model.

9. Compute the response variable y using the linear model equation $y = \beta x^T + \epsilon$.
10. (Validate the generated data by calculating the empirical R^2 value.)

The generated data set Z includes the response variable and the predictor variable. We subtract 1 for the first predictor and add $\epsilon + 10$ to generate outliers. This results in moderate outliers seperated but not far from the main bulk of the regular observations. The missing mechanism for setting missings to the response variable is for all simulations MAR with selection probabilities proportional to the magnitude of values in the first predictor variable.

For simulations II and III the number of observations, the correlation structure, the number of missing values, and the number of outliers were varied. Figure 2 shows only the result of one parameter setting, while the results for other 23 settings are included in the supplementary file.

There exists a multitude of data sets and configurations that could be selected for this analysis. The current selection represents a scenario where outliers are in close proximity to the majority of the data set. It is also feasible to alter the rates of missing values and their respective positions. However, it is important to emphasize—as illustrated in Fig. 8 and Table 3—that the GAMLSS imputation methods can be quite resource-intensive in terms of time. The number of replications is 100 for precision measures and 250 for coverage rates.

4 Results

4.1 Visual comparison of a multiple imputation

We previously discussed Fig. 1 in the context of GAMLSS NO and GAMLSS TF methods. It shows multiple imputations of two data points with missing value in the response variable NOx. This figure should first motivate the newly proposed imputation methods compared with existing GAM-based imputation methods. We observe that the scatter of multiple imputations using *gam* or *gamRob* is generally satisfactory across all methods. An exception is the dispersion for robust bootstrap, which uses weights from the Bacon algorithm, which appears overly constrained. The *gam* imputations exhibit bias—the average of the multiple imputations falls outside the range of the observed data, while the imputation mean from *gamRob* seems unbiased. It is evident that all of these methods greatly exceed the performance of GAMLSS. GAMLSS imputations exhibit excessive variability, and their mean does not align with the observed data points.

4.2 Visual comparison of a single imputation

Next, we also visually compare our methods with GAMLSS imputation for the ethanol data, including the outliers visible in Fig. 2. The main focus of this figure is (1) to look at some first obvious differences between the methods, to look at their robustness in a rough and initial way, and in particular to compare different approaches for imputation uncertainty (see Algorithm 6). GAMLSS with normal distribution (NO) performs similarly to the assumed t-distribution (TF). On first sight, those outcomes look quite robust, which is not the case by the use of non-robust GAM within the imputeRobust algorithm. This leads to the worst results, in general, because the imputed values are affected by outliers. Robust fitting using gamRob (and default robust bootstrap) fixes this, and imputations are no longer driven by outliers. The proposed *gamRob* algorithm from the imputeRobust algorithm works best to impute the ethanol data (based on one realization), whereby the PMM imputation uncertainties is superior.

4.3 Results on precision measures

In the following, a selection of possible variations of methods is chosen, such as different bootstrap approaches (quantile, Bacon, stratified) and methods for imputation uncertainty (PMM and midastouch).

Figure 3a and b show the NRMSE's and MSECOR's for the ethanol data. The most important results, among many details, are:

- Imputation with the proposed robust GAM using the weights from the Bacon algorithm for the robust bootstrap performs best.
- The imputation uncertainty considered with midastouch does not work well for nor-robust GAM.
- 10% of the imputations with GAMLSS TF were not successful, i.e., GAMLSS TF produced an error. In 4% of the cases the imputations were completely *wrong*, i.e. much too high values were imputed.
- In 13% GAMLSS NO produced an error to impute a data set, and once very unrealistic values were imputed.

Figure 4a and b show the NRMSE's from the dolphin data set. The most important results, among many details, are:

- Classical GAM with robust bootstrap by quantile or stratified and PMM for imputation uncertainty outperforms other methods.
- Robust GAM with robust stratified bootstrap leads to the second-best results.
- GAMLSS TF and GAMLSS NO mostly produced errors (in 78% and 81% of the data sets), therefore they are not

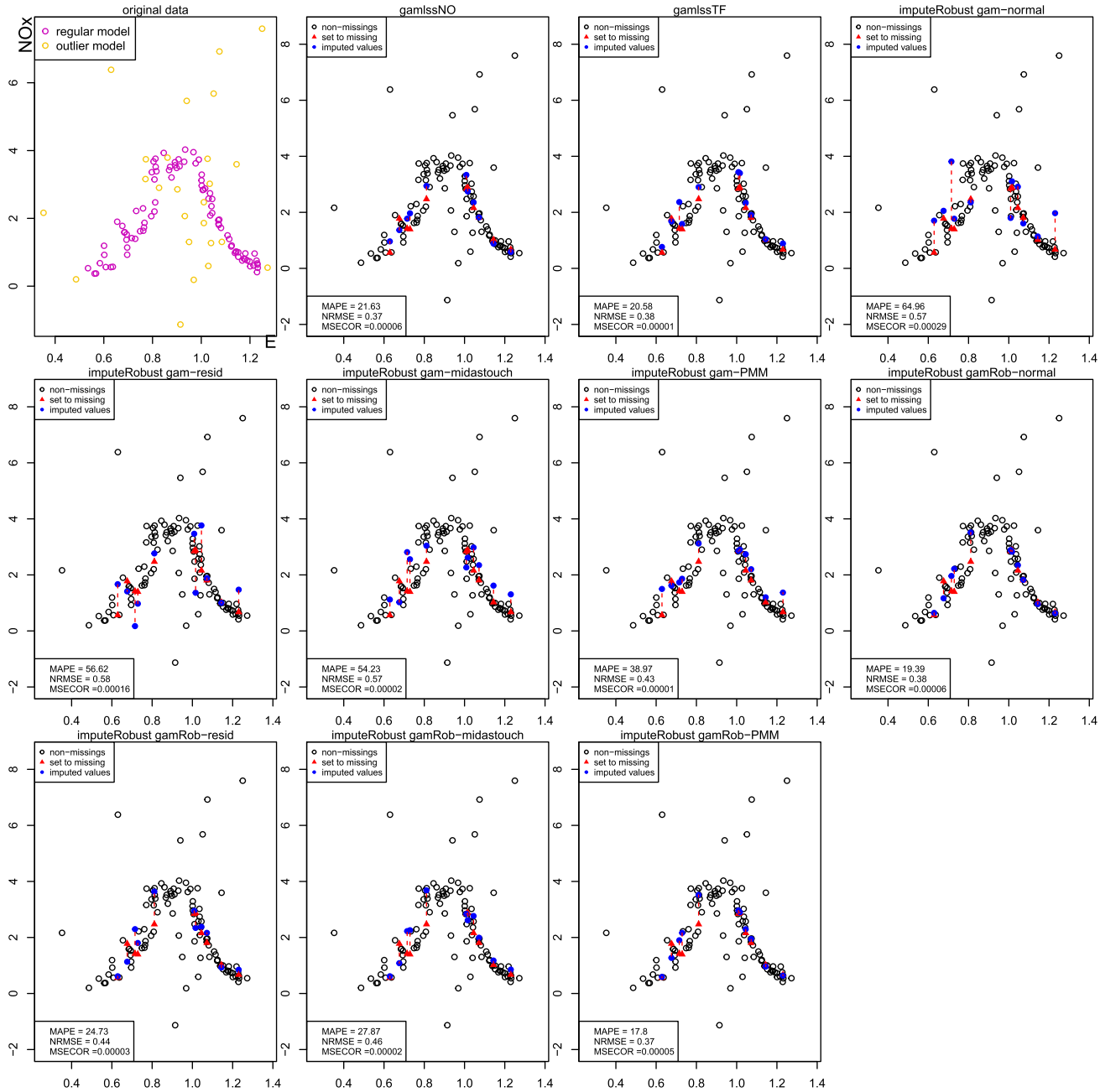


Fig. 2 Imputation (in blue) of missing values (in red) of the ethanol data set including outliers (in orange) (colour figure online)

really comparable to other methods because the remaining results might be too overoptimistic.

- Robust GAM with robust stratitifed bootstrap outperforms robust GAM with robust bootstrap based on weights from the Bacon algorithm.

Classical GAM with robust bootstrap by quantile or stratified and PMM for imputation uncertainty outperforms other methods, followed by robust GAM with stratified bootstrap and GAMLSS NO. However, GAMLSS NO once imputed very badly and failed by 13 out of 100 imputations (GAMLSSTF failed on 18). Generally, PMM is slightly preferable to midastouch, and robust bootstrap with weights from the Bacon algorithm gives higher NRMSE values. A very similar picture can be seen in Fig. 4b. GAMLSSTF failed often, and thus the remaining results might be too overoptimistic.

A different picture is obtained by introducing outliers to the dolphin data set; see Fig. 5a and b. The most important results, among many details, are:

- The proposed robust GAM method performs best for the MSECOR measure, whereas the best results regarding the NRMSE are obtained with GAM with imputation uncertainty PMM.
- Again, robust bootstrap based on weights from the Bacon algorithm is worse than robust stratified bootstrap for the robust GAM.
- In 85% and 82% of the imputations, GAMLSS TF and GAMLSS NO produced errors. Successful imputations are much worse than with all other methods.

Figure 6a and b show the evaluation of the imputations of the simulated data. The most important results, among many details, are as follows:

- The results are similar to those obtained from the imputation of the ethanol data set.
- GAM with PMM as the imputation uncertainty method provides the best results.
- GAMLSS TF and GAMLSS NO give the worst results for NRMSE.

Figure 7a and b show the evaluation of the imputations of the simulated data with outliers. The most important results, among many details, are:

- The results are similar to those for the imputation of the ethanol data set and the simulated data set without outliers.

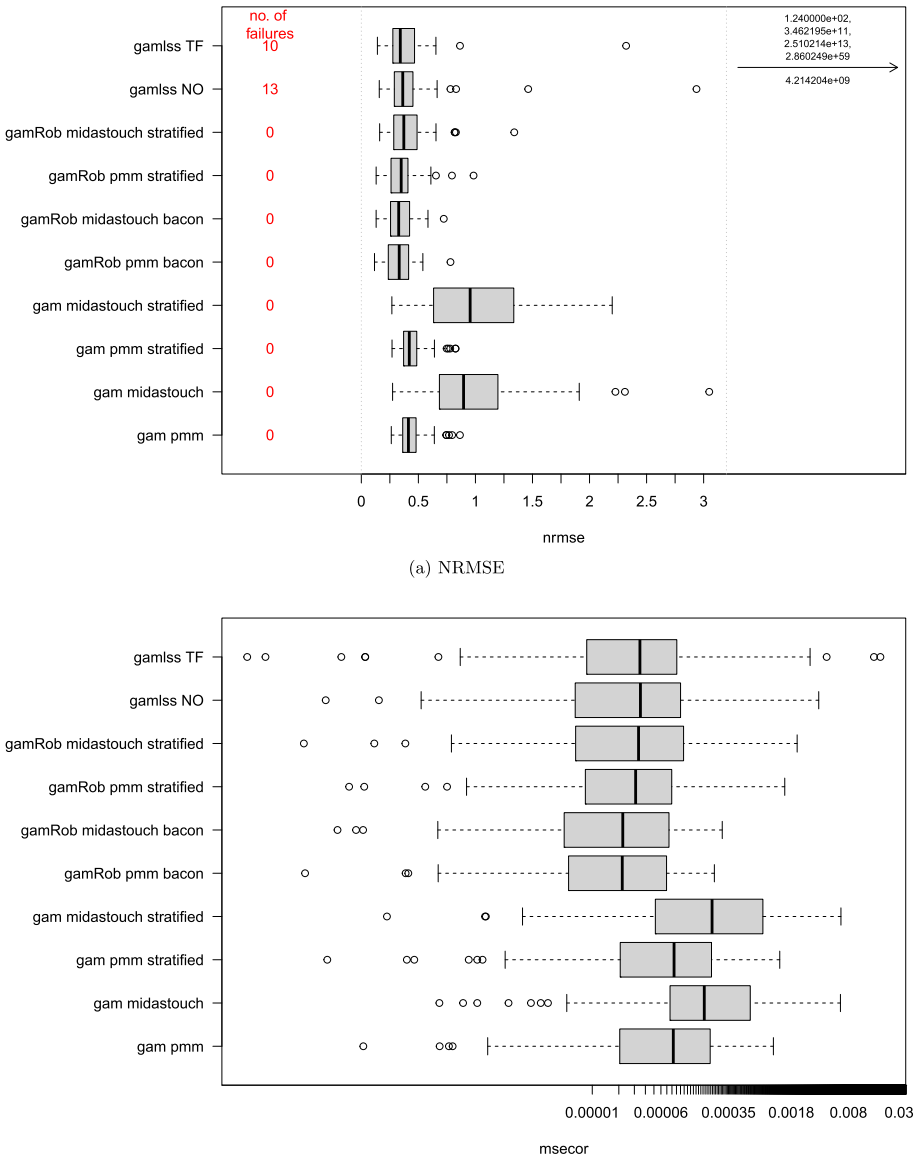
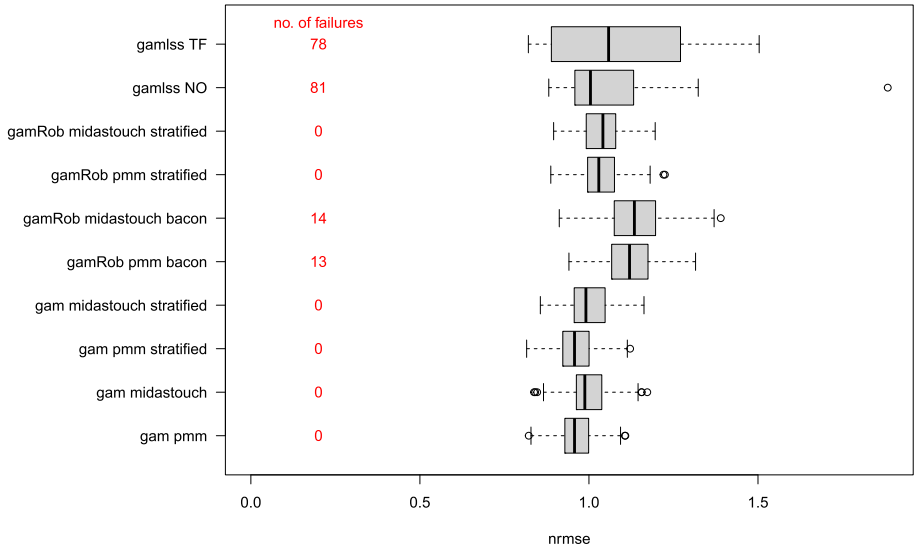


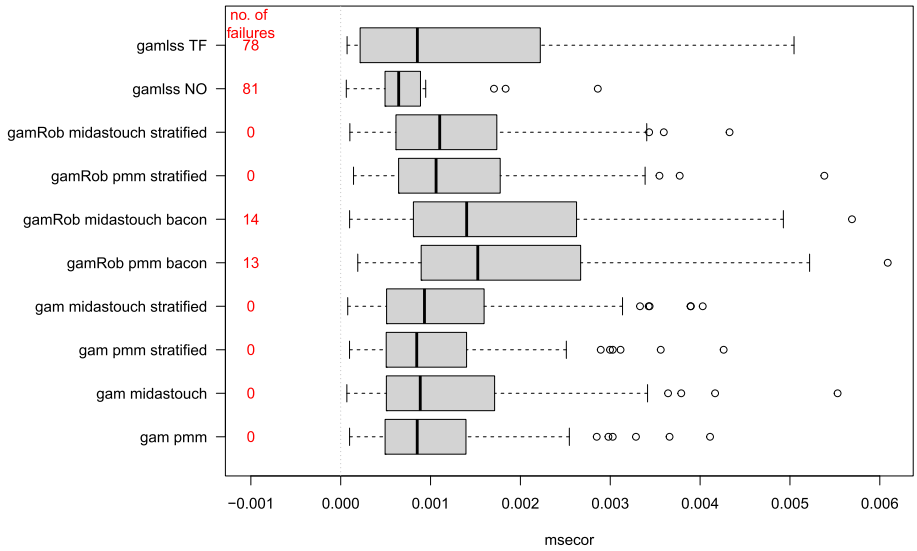
Fig. 3 Imputations of the ethanol data set

(a) NRMSE

(b) MSECOR in log scale.



(a) NRMSE



(b) MSECOR in log scale.

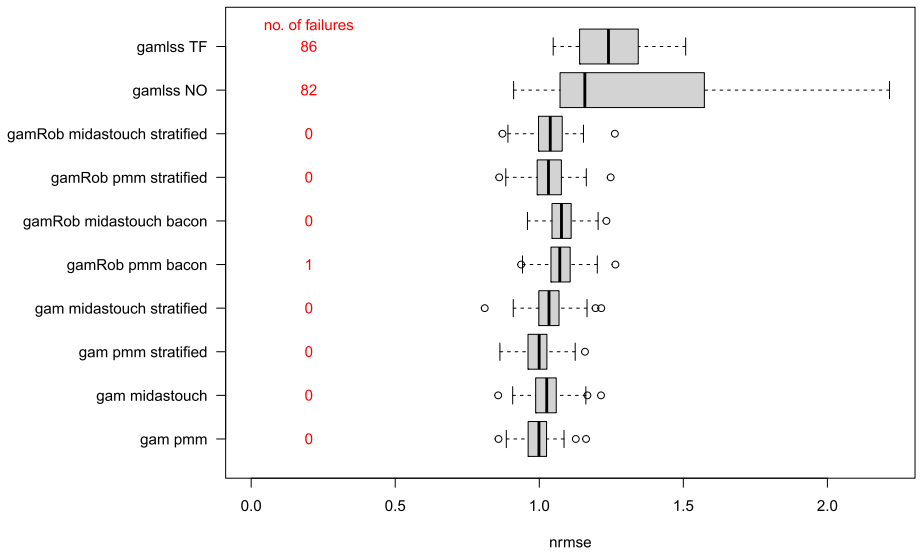
Fig. 4 Imputations of the dolphin data set

- GAMLSS TF and GAMLSS NO produce the worst results for NRMSE and MSECOR and also produce errors 13% and 12% of the time.

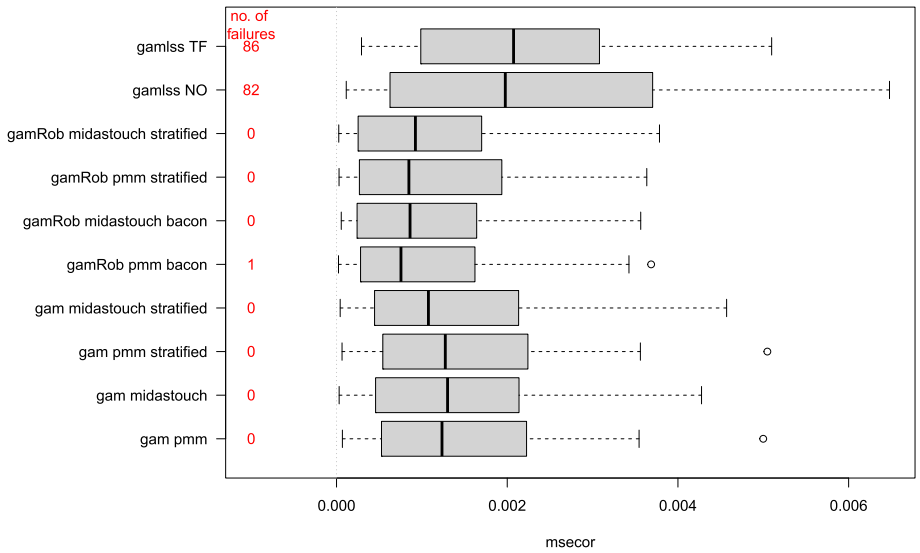
4.4 Coverage rates and root mean squared errors

Table 1 shows the coverage rates and root mean squared errors for the simulated data (setting I) with and without outliers. The coverage rates and the root mean squared errors are estimated for the arithmetic mean of the target variable. For the simulation with outliers, the arithmetic mean is estimated for the non-outliers. Note that only in the non-outliers part missing values are introduced. For comparison reasons, we also show coverage rates of other imputation methods mice's PMM and mice's midastouch default implementation as well as for kNN and IRMI as well as for missRanger in the supplementary material.

It can be seen that the coverage rates decrease a bit when data are simulated with outliers. Note that compared to the standard predictive mean matching of mice and also of the midastouch approach, which is all based on linear regression models in our case, are dropping significantly as soon as outliers are present in the data (see supplementary material). This is not the case with other methods, apart from the GAMLSS methods. These provide comparably good coverage rates and mean square errors, but fail when outliers are present in the data. With GAMLSS-TF, not only one imputation was successful, and even with the correction of extreme values—a feature of the impute-robust imputation of GAMLSS—some of the values imputed with GAMLSS-NO are very extreme.



(a) NRMSE



(b) MSECOR in log scale.

Fig. 5 Imputations of the dolphin data set including outliers

Table 2 shows the result of one parameter setting for simulation experiments II and III. Please be informed that additional outcomes for 23 different parameter configurations can be found in the supplementary material. Only GAMLSS based imputation methods fail to have acceptable coverage rates for setting III on the regression coefficients.

When comparing to the additional results in the supplementary file, coverage rates are always reasonable for robust GAM imputation at least when the outlier rate is moderate and the rate of missing values is not very high, i.e. realistic for practical real-world settings.

4.5 Empirical computation time evaluation

Figure 8 and Table 3 summarize the computational times. Both GAMLSS imputation procedures took in the range of 174 and 290 times higher computational time than with imputation using GAM with PMM as imputation uncertainty method and robust bootstrap based on quantiles. The robust GAM methods need about 2–5.4 times longer to compute than the non-robust versions.

5 Conclusion

In summary, the application of robust imputation methods grounded in Generalized Additive Models (GAMs) com-

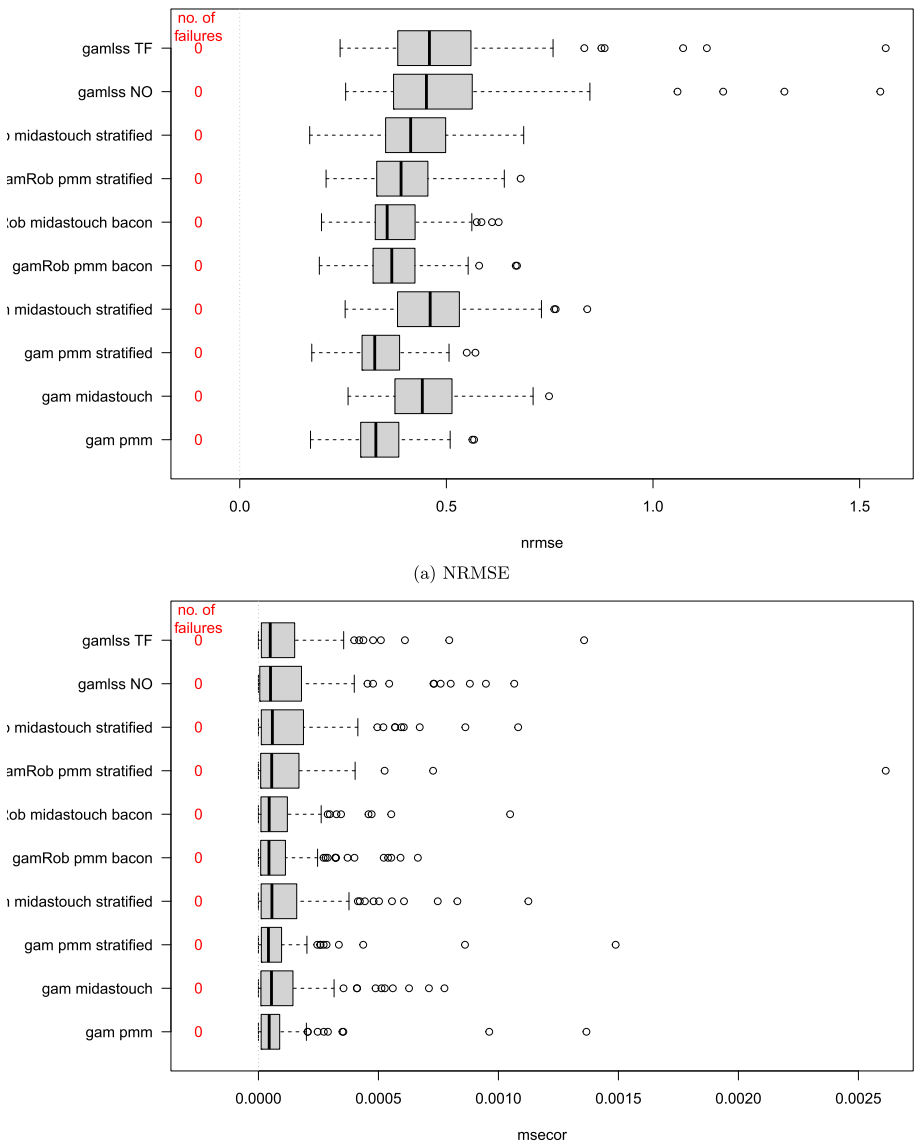


Fig. 6 Imputations of the simulated data

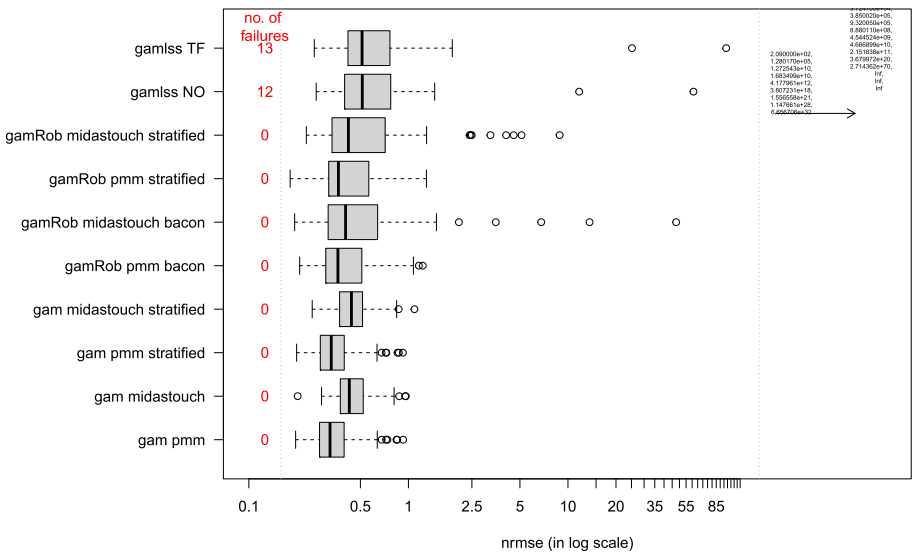
(a) NRMSE

(b) MSECOR in log scale.

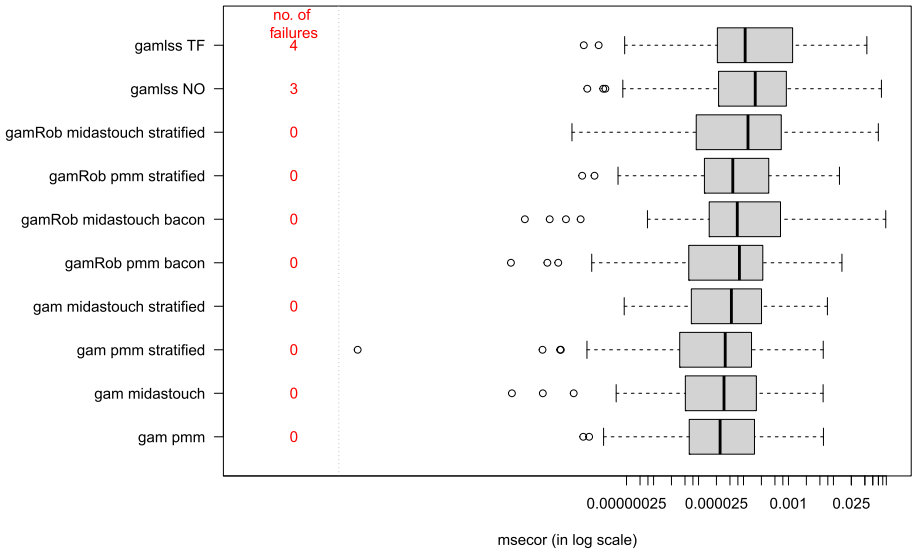
Table 1 Coverage rates and root mean squared errors of the arithmetic mean of the target variable for the simulated data and simulated data with outliers

Data				Simulated		Simulated with outliers	
Method	R package	Imputation uncertainty	Model uncert./bootstrap	CR	RMSE	CR	RMSE
ImputeRobust/GAM	VIM*	PMM	Robust-quantile	0.88	0.25	0.72	0.35
ImputeRobust/GAM	VIM*	midastouch	Robust-quantile	0.90	0.25	0.72	0.37
ImputeRobust/GAM	VIM*	PMM	Robust-stratified	0.89	0.25	0.76	0.37
ImputeRobust/GAM	VIM*	midastouch	Robust-stratified	0.90	0.25	0.75	0.37
ImputeRobust/robGAM	VIM*	PMM	Robust-Bacon	0.83	0.25	0.74	0.36
ImputeRobust/robGAM	VIM*	midastouch	Robust-Bacon	0.83	0.26	0.76	0.33
ImputeRobust/robGAM	VIM*	PMM	Robust-stratified	0.87	0.26	0.74	0.35
ImputeRobust/robGAM	VIM*	midastouch	Robust-stratified	0.91	0.25	0.77	0.34
gamlss-NO	ImputeRobust	PMM	Classical	0.89	0.26	0.86	$2.04e^{62}$
gamlss-TF	ImputeRobust	PMM	Classical	0.92	0.25	–	–

VIM*: available in the development version of VIM at GitHub, https://github.com/statistikat/VIM



(a) NRMSE



(b) MSECOR in log scale.

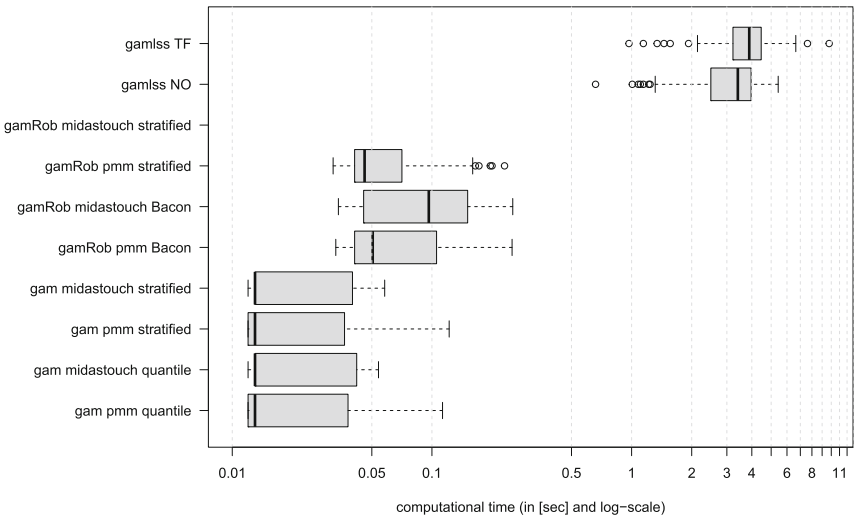
Fig. 7 Imputations of the simulated data including outliers

bined with a robust bootstrap procedure offers significant advantages over other imputation techniques whenever the dependencies between variables cannot be linearized. It leads to significantly better imputation results compared to the GAMLSS imputation framework.

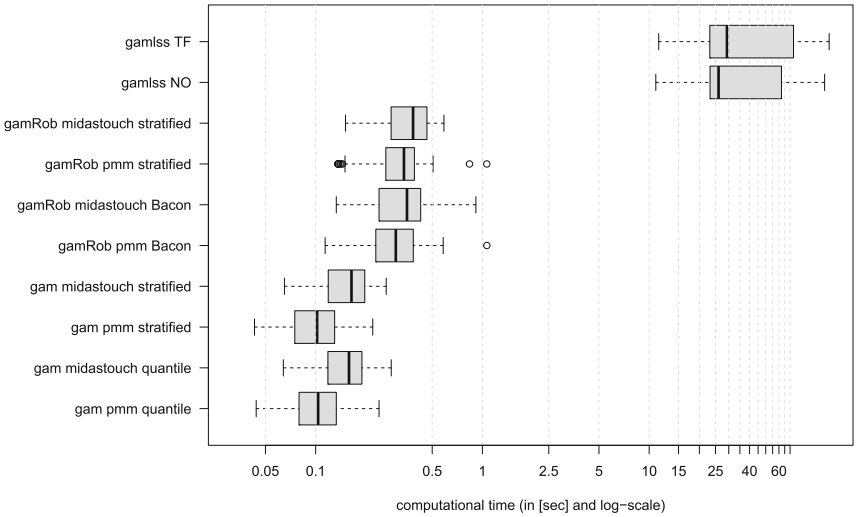
Firstly, GAMs provide the flexibility to model complex, nonlinear relationships between variables, which are common in real-world data. This flexibility ensures that the imputed values are more accurate and reflective of the underlying potential nonlinear data structure, as opposed to the assumption of linearity required by many other methods. However, it has been found that using GAMLSS to impute missing values is sensitive to outliers, can lead to poorer imputations or errors (especially if outliers are present in the data), and computation times are more than 75 times higher compared to our proposed robust imputation method. Even for data without outliers, the methods using GAMLSS imputation methods are usually outperformed by our proposed methods. Moreover, GAMLSS often fails to impute missing values, for example, in about 80% of cases for the dolphin data set.

Secondly, the robustness of the method to outliers and non-normal data distribution is crucial, as traditional imputation methods can be sensitive to such anomalies, leading to biased imputations. Our proposed robust version of GAM using the Bacon algorithm mitigates this issue by using procedures that are less affected by extremes in the data, thereby maintaining the integrity of the data's core characteristics during the imputation process. In addition, the Bacon algorithm is computationally efficient and is therefore suitable for multiple imputation with sequential conditional imputation.

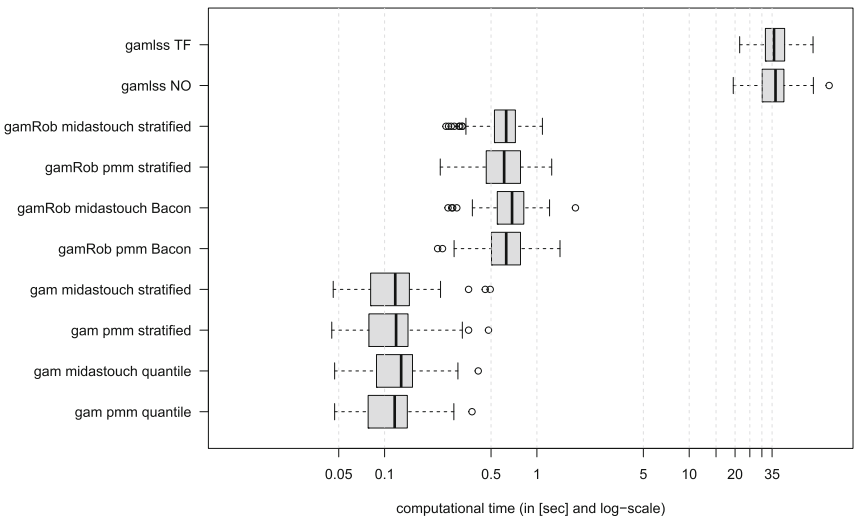
Fig. 8 Computational times [in seconds]



(a) Ethanol data set.



(b) dolphin data set.



(c) Simulated data set.

Table 2 Simulation results with the following settings: n = 200; missing rate = 0.1; correlation = 0.5; outlier rate = 0

Kind	Method	R package	Imputation uncert	Bootstrap	CR	RMSE
μ_1	ImputeRobust/robGAM	VIM*	PMM	Robust-Bacon	0.94	0.06
μ_1	ImputeRobust/robGAM	VIM*	midastouch	Robust-Bacon	0.96	0.06
μ_1	ImputeRobust/robGAM	VIM*	PMM	Robust-stratified	0.94	0.06
μ_1	ImputeRobust/robGAM	VIM*	midastouch	Robust-stratified	0.96	0.06
μ_1	gamlss-NO	ImputeRobust	PMM	Classical	0.94	0.06
μ_1	gamlss-TF	ImputeRobust	PMM	Classical	0.94	0.06
r_{12}	ImputeRobust/robGAM	VIM*	PMM	Robust-Bacon	0.94	0.06
r_{12}	ImputeRobust/robGAM	VIM*	midastouch	Robust-Bacon	0.94	0.06
r_{12}	ImputeRobust/robGAM	VIM*	PMM	Robust-stratified	0.94	0.06
r_{12}	ImputeRobust/robGAM	VIM*	midastouch	Robust-stratified	0.92	0.06
r_{12}	gamlss-NO	ImputeRobust	PMM	Classical	0.96	0.05
r_{12}	gamlss-TF	ImputeRobust	PMM	Classical	0.94	0.05
β_1	ImputeRobust/robGAM	VIM*	PMM	Robust-Bacon	0.88	0.48
β_1	ImputeRobust/robGAM	VIM*	midastouch	Robust-Bacon	0.94	0.48
β_1	ImputeRobust/robGAM	VIM*	PMM	Robust-stratified	0.90	0.48
β_1	ImputeRobust/robGAM	VIM*	midastouch	Robust-stratified	0.90	0.48
β_1	gamlss-NO	ImputeRobust	PMM	Classical	0.48	0.27
β_1	gamlss-TF	ImputeRobust	PMM	Classical	0.46	0.28

Kinds μ_1 and r_{12} are related to simulation II, and the estimators of interest are the variance of the arithmetic mean of the first variable (mean) and the variance of the correlation between the first two variables (cor). The regression parameter β_1 corresponds to simulation III, and investigates the variance of the regression slope of the first predictor

Third, the robust bootstrap technique enhances the method by providing a measure of variability and uncertainty in the imputation process that accounts for model uncertainty. This stochastic element—together with considering imputation uncertainty using PMM or midastouch—enables the generation of multiple imputed data sets, accounting for the inherent randomness. It allows for better statistical inference by reflecting the natural variability in the data and offering a range of plausible values instead of a single point estimate.

Fourth, in particular, for our robust imputation approach, robust bootstrap, where only the last good subset of the Bacon algorithm is selected to sample, gave the best results for the ethanol data, but not always for the other datasets. PMM is preferable to midastouch (and other noise strategies such as normal noise), since for nonlinear relationships, it makes sense to only consider close neighbors in the predictor space. The robust version of the GAM with the Bacon algorithm was sometimes better than using the non-robust GAM. In general, the more and the larger the outliers are, the more reliable the robust version is.

Fifth, compared to other methods, the robust GAM-based approach with robust bootstrap also contributes to the reliability of subsequent statistical analyzes. By providing more reliable imputed values, the risk of model misspecification and type I and type II errors in hypothesis testing is reduced. This contributes to more robust conclusions and can enhance the validity of research findings or decision-making processes based on the imputed data set(s).

Sixth, coverage rates were *only* mostly around 0.9–0.96, sometimes a bit lower which is to be almost critical according to Schafer and Graham (2002). However, we like to mention that coverage rates lower dramatically for non-robust imputation methods when outliers are present, see supplementary material where for standard multiple imputation methods such as the default implementation of mice the coverage rates drop to 0.4, because outliers influence the imputation model. Note that the coverage rates of GAMLSS are sometimes very low. It is anticipated that the coverage rates of our robust imputation method will fall within an acceptable range, as our proposed techniques uses PMM (or midastouch) which are well-known methods for imputation uncertainty, and bootstrapping—even adapted for robustness—for model uncertainty, and downweighting of outliers should not influence coverage rates because they are calculated based on non-outliers. Note that in practice, coverage rates may often be "close to" 0.95 but not exactly 0.95, for example, because only a limited (relatively) small number of multiple imputations are carried out and many processes influence them like the complexity of GAM and its smoothing.

Finally, encompassing both robustness in estimation and randomness in real-world data application and imputation, this method stands out as particularly suited for practical applications where data quality cannot be guaranteed and

Table 3 Computational time (in seconds) from 100 runs each

Method	Data	*Mean*	*Median*	*Ratio*	*sd*	*mad*
gam pmm quantile	dolphin	0.11	0.10	1.0	0.04	0.044
gam midastouch quantile	dolphin	0.15	0.16	1.6	0.05	0.048
gam pmm stratified	dolphin	0.10	0.10	1.0	0.04	0.042
gam midastouch stratified	dolphin	0.16	0.16	1.6	0.06	0.056
gamRob pmm Bacon	dolphin	0.32	0.30	3.0	0.13	0.118
gamRob midastouch Bacon	dolphin	0.35	0.35	3.5	0.13	0.125
gamRob pmm stratified	dolphin	0.33	0.34	3.3	0.12	0.088
gamRob midastouch stratified	dolphin	0.37	0.38	3.8	0.11	0.123
gamlss NO	dolphin	41.54	26.07	255.6	28.59	6.611
gamlss TF	dolphin	44.65	29.28	287.1	30.40	12.022
gam pmm quantile	dolphin_out	0.12	0.11	1.0	0.05	0.044
gam midastouch quantile	dolphin_out	0.18	0.18	1.7	0.07	0.066
gam pmm stratified	dolphin_out	0.12	0.12	1.1	0.05	0.054
gam midastouch stratified	dolphin_out	0.18	0.19	1.7	0.07	0.069
gamRob pmm Bacon	dolphin_out	0.36	0.35	3.2	0.17	0.133
gamRob midastouch Bacon	dolphin_out	0.41	0.41	3.7	0.17	0.139
gamRob pmm stratified	dolphin_out	0.41	0.37	3.4	0.19	0.179
gamRob midastouch stratified1	dolphin_out	0.44	0.41	3.7	0.22	0.145
gamlss NO	dolphin_out	48.60	31.91	290.1	36.87	10.601
gamlss TF	dolphin_out	46.95	30.80	280.0	34.44	8.168
gam pmm quantile	Ethanol	0.02	0.01	1	0.02	0.001
gam midastouch quantile	Ethanol	0.02	0.01	1.7	0.01	0.001
gam pmm stratified	Ethanol	0.02	0.01	1.0	0.02	0.001
gam midastouch stratified	Ethanol	0.02	0.01	1.7	0.01	0.000
gamRob pmm Bacon	Ethanol	0.08	0.05	3.1	0.05	0.018
gamRob midastouch Bacon	Ethanol	0.10	0.10	3.6	0.06	0.076
gamRob pmm stratified	Ethanol	0.07	0.05	3.4	0.04	0.009
gamRob midastouch stratified	Ethanol	0.00	0.00	3.7	0.00	0.000
gamlss NO	Ethanol	3.24	3.40	290.2	1.08	1.054
gamlss TF	Ethanol	3.90	3.88	279.1	1.23	0.945
gam pmm quantile	gamSim_out	0.12	0.11	1.0	0.10	0.039
gam midastouch quantile	gamSim_out	0.12	0.12	1.1	0.04	0.040
gam pmm stratified	gamSim_out	0.11	0.11	1.0	0.04	0.035
gam midastouch stratified	gamSim_out	0.12	0.12	1.1	0.04	0.039
gamRob pmm Bacon	gamSim_out	0.60	0.60	5.4	0.21	0.182
gamRob midastouch Bacon	gamSim_out	0.60	0.59	5.3	0.18	0.158
gamRob pmm stratified	gamSim_out	0.58	0.58	5.2	0.17	0.181
gamRob midastouch stratified	gamSim_out	0.59	0.57	5.1	0.17	0.153
gamlss NO	gamSim_out	19.47	19.44	174.3	3.58	3.173
gamlss TF	gamSim_out	22.23	22.11	198.3	4.22	3.799

The ratio of the average computation time determines the relative speed. A ratio > 1.0 (< 1.0) implies that a method is faster (slower) than non-robust GAM impuation using PMM and quantiles for the robust bootstrap

the costs of poor imputation are high. It is a strong methodology that leverages advanced statistical techniques to deliver high-quality imputations of data with nonlinear relationships, ensuring that analyses are based on data that closely resemble true, unobserved values.

Future work: Categorical variables were present and used; for example, in the imputation of the dolphin dataset four categorical variables were used, but we did not set missing in categorical variables and impute categorical variables. Categorical variables were not included in the simulation study and in the additional simulation studies in the supplementary material to keep the simulations straight. Naturally, note that, in general, categorical predictor variables are not smoothed in GAM at all. We have not solved the problem of imputing missing values for categorical variables, as this would require a very different robust approach and is beyond the scope of this paper, as typical robust procedures such as our modified Bacon algorithm will not work. Robust GAM for categorical responses is a topic for future research. With the chain in our proposed methods, non-robust methods can be integrated to impute missing values in categorical variables.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11222-024-10429-1.

Author Contributions Since there is only one author, this does not apply.

Funding Open access funding provided by FHNW University of Applied Sciences and Arts Northwestern Switzerland

Availability of data and materials All data are from public sources, well described in the article.

Declarations

Funding No financial support have been provided.

Conflict of interest There are no financial or non-financial conflicts of interest

Ethics approval No study-specific approval from the appropriate ethics committee is required.

Consent to participate Not applicable.

Consent for publication Not applicable.

Code availability Upon request.

References

Bartlett, J.W., Hughes, R.A.: Bootstrap inference for multiple imputation under uncongeniality and misspecification. Stat. Methods Med. Res. **29**(12), 3533–3546 (2020). https://doi.org/10.1177/0962280220932189

Beaton, A.E., Tukey, J.W.: The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data. Technometrics **16**(2), 147–185 (1974). https://doi.org/10.1080/00401706.1974.10489171

Billor, N., Hadi, A.S., Velleman, P.F.: BACON: blocked adaptive computationally efficient outlier nominators. Comput. Stat. Data Anal. **34**(3), 279–298 (2000). https://doi.org/10.1016/S0167-9473(99)00101-2

Brinkman, N.D.: Ethanol fuel-a single-cylinder engine study of efficiency and exhaust emissions. In: SAE International Congress and Exposition. SAE International (1981). https://doi.org/10.4271/810345

Brownlee, J.: Data Preparation for Machine Learning: Data Cleaning, Feature Selection, and Data Transforms in Python. Machine Learning Mastery, San Francisco (2020)

Buuren, S.: Flexible Imputation of Missing Data. Chapman & Hall/CRC Interdisciplinary Statistics. Taylor & Francis, Boca Raton (2012). https://doi.org/10.1201/9780429492259

Cédric, B., Beat, H.: The BACON-EEM algorithm for multivariate outlier detection in incomplete survey data. Surv. Methodol. Stat. Can. 34, 91–103 (2008)

Chambers, R.L.: Outlier robust finite population estimation. J. Am. Stat. Assoc. **81**, 1063–1069 (1986). https://doi.org/10.1080/01621459.1986.10478374

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R.: Crisp-dm 1.0 step-by-step data mining guide. Technical report, The CRISP-DM consortium (2000)

Filzmoser, P., Gregorich, M.: Multivariate outlier detection in applied data analysis: global, local, compositional and cellwise outliers. Math. Geosci. **52**(8), 1049–1066 (2020). https://doi.org/10.1007/s11004-020-09861-6

Grentzelos, C., Caroni, C., Barranco-Chamorro, I.: A comparative study of methods to handle outliers in multivariate data analysis. Comput. Math. Methods **3**(3), 1129 (2021). https://doi.org/10.1002/cmm4.1129

Gu, C., Wahba, G.: Minimizing GCV/GML scores with multiple smoothing parameters via the newton method. SIAM J. Sci. Stat. Comput. **12**(2), 383–398 (1991). https://doi.org/10.1137/0912021

Hippel, P.T., Bartlett, J.W.: Maximum likelihood multiple imputation: faster imputations and consistent standard errors without posterior draws. Stat. Sci. **36**(3), 400–420 (2021). https://doi.org/10.1214/20-STS793

Honaker, J., King, G.: What to do about missing values in time-series cross-section data. Am. J. Polit. Sci. **54**(2), 561–581 (2010). https://doi.org/10.1111/j.1540-5907.2010.00447.x

Honaker, J., King, G., Blackwell, M.: Amelia II: a program for missing data. J. Stat. Softw. **45**(7), 1–47 (2011). https://doi.org/10.18637/jss.v045.i07

Jong, S.V.B., Spiess, M.: Multiple imputation of predictor variables using generalized additive models. Commun. Stat. Simul. Comput. **45**(3), 968–985 (2016). https://doi.org/10.1080/03610918.2014.911894

Kowarik, A., Templ, M.: Imputation with the R package VIM. J. Stat. Softw. **74**(7), 1–16 (2016). https://doi.org/10.18637/jss.v074.i07

Little, R.J.A., Rubin, D.B.: Statistical Analysis with Missing Data. Wiley Series in Probability and Mathematical Statistics. Probability and Mathematical Statistics. Wiley, New York (2002). http://books.google.com/books?id=aYPwAAAAMAAJ

Loo, M., Jonge, E.: Statistical Data Cleaning with Applications in R. Wiley, New York (2018)

Mavrogiorgou, A., Kiourtis, A., Manias, G., Kyriazis, D.: Adjustable data cleaning towards extracting statistical information. Stud. Health Technol. Inform. **281**, 1013–1014 (2021). https://doi.org/10.3233/SHTI210332

Meng, X.-L.: Multiple-imputation inferences with uncongenial sources of input. Stat. Sci. **9**(4), 538–558 (1994). https://doi.org/10.1214/ss/1177010269

Piwetz, S.: Common bottlenose dolphin (tursiops truncatus) behavior in an active narrow seaport. PLoS ONE **14**(2), 1–23 (2019). https://doi.org/10.1371/journal.pone.0211971

Rahm, E., Do, H.H.: Data cleaning: problems and current approaches. IEEE Data Eng. Bull. **23**, 3–13 (2000)

Rigby, R.A., Stasinopoulos, D.M.: Generalized additive models for location, scale and shape. J. R. Stat. Soc. Ser. C (Appl. Stat.) **54**(3), 507–554 (2005). https://doi.org/10.1111/j.1467-9876.2005.00510.x

Salfran, D., Spiess, M.: The R journal: generalized additive model multiple imputation by chained equations with package imputerobust. R J. **10**, 61–72 (2018). https://doi.org/10.32614/RJ-2018-014

Salibián-Barrera, M., Van Aelst, S., Willems, G.: Fast and robust bootstrap. Stat. Methods Appl. **17**(1), 41–71 (2008). https://doi.org/10.1007/s10260-007-0048-6

Schafer, J.L., Graham, J.W.: Missing data: our view of the state of the art. Psychol. Methods **7**(2), 147–177 (2002)

Schoch, T.: wbacon: Weighted BACON algorithms for multivariate outlier nomination (detection) and robust linear regression. J. Open Source Softw. 6(62), 3238 (2021). https://doi.org/10.21105/joss.03238 https://doi.org/10.21105/joss.03238

Shao, J., Sitter, R.R.: Bootstrap for imputed survey data. J. Am. Stat. Assoc. **91**(435), 1278–1288 (1996). https://doi.org/10.1080/01621459.1996.10476997

Stasinopoulos, D., Rigby, R., Heller, G., Voudouris, V., De Bastiani, F.: Flexible Regression and Smoothing: Using GAMLSS in R. Chapman and Hall/CRC the R Series. Chapman & Hall, London (2017). https://doi.org/10.1201/b21973

Stekhoven, D.J., Bühlmann, P.: MissForest—non-parametric missing value imputation for mixed-type data. Bioinformatics **28**(1), 112–118 (2011). https://doi.org/10.1093/bioinformatics/btr597

Templ, M.: Imputation and Visualization of Missing Values, p. 561. Springer, Cham (2023). (**in print**)

Templ, M., Kowarik, A., Filzmoser, P.: Iterative stepwise regression imputation using standard and robust methods. Comput. Stat. Data Anal. **55**(10), 2793–2806 (2011). https://doi.org/10.1016/j.csda.2011.04.012

Templ, M., Gussenbauer, J., Filzmoser, P.: Evaluation of robust outlier detection methods for zero-inflated complex data. J. Appl. Stat. **0**(0), 1–24 (2019). https://doi.org/10.1080/02664763.2019.1671961

Todorov, V., Templ, M., Filzmoser, P.: Detection of multivariate outliers in business survey data with incomplete information. Adv. Data Anal. Classif. **5**(1), 37–56 (2011)

Vale, S.: Generic Statistical Business Process Model. Joint UNECE/Eurostat/OECD Work Session on Statistical Metadata (METIS) (2009)

Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S, 4th edn. Springer, New York (2002). https://doi.org/10.1007/978-0-387-21706-2

Wood, S.N.: Thin plate regression splines. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **65**(1), 95–114 (2003). https://doi.org/10.1111/1467-9868.00374

Wood, S.: Generalized Additive Models: An Introduction with R. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, New York (2006)