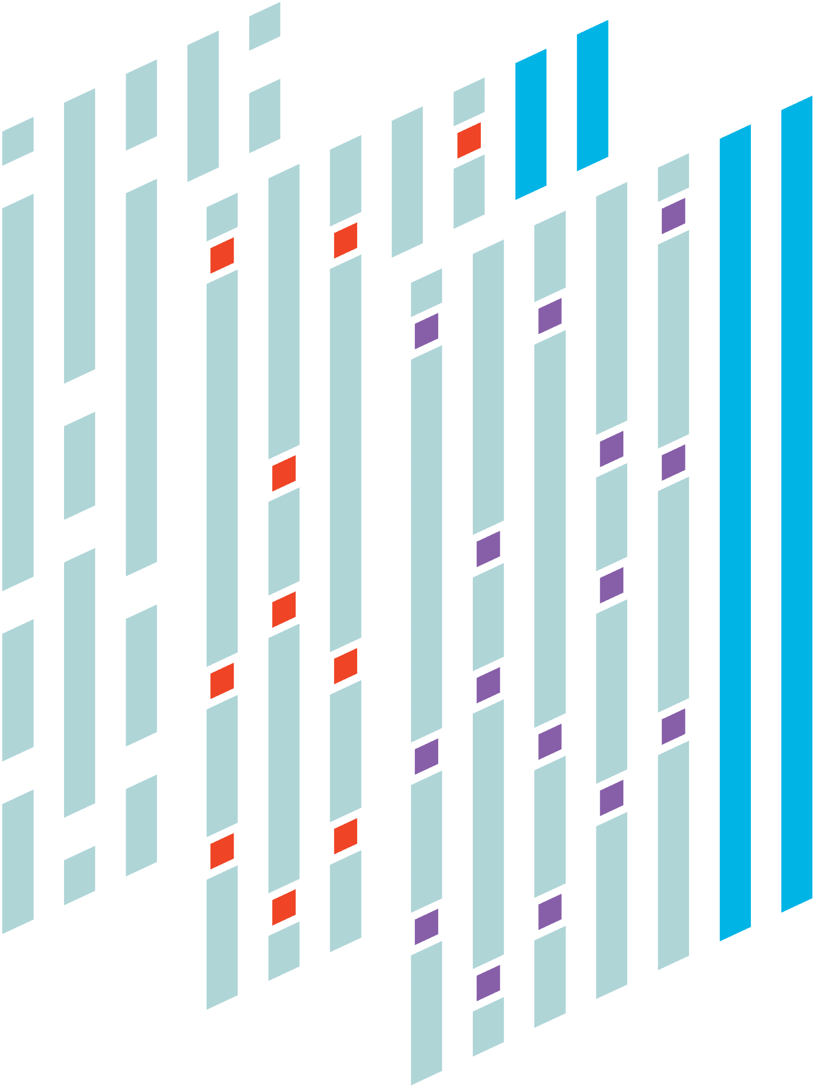
TILBURG ◆ UNIVERSITY

Simplifying imputation with many predictors in MICE using principal component analysis

Authors	Costantini,E.
DOI	10.26116/tsb.30823607
Publication Date	2024-10-18
Document Version	publishersversion
Link	https://research.tilburguniversity.edu/en/publications/4b203e1c-463d-4097-ae43-2d58ef269446
Citation	Costantini, E 2024, 'Simplifying imputation with many predictors in MICE using principal component analysis', Doctor of Philosophy, s.l.. https://doi.org/10.26116/tsb.30823607
Download Date	
Rights	

SIMPLIFYING IMPUTATION WITH MANY PREDICTORS IN MICE USING PRINCIPAL COMPONENT ANALYSIS



EDOARDO COSTANTINI

Simplifying Imputation with Many Predictors in MICE Using Principal Component Analysis

Proefschrift ter verkrijging van de graad van doctor aan Tilburg University
op gezag van de rector magnificus, prof. dr. W.B.H.J. van de Donk, in het
openbaar te verdedigen ten overstaan van een door het college voor
promoties aangewezen commissie in de Aula van de Universiteit op
vrijdag 18 oktober 2024 om 10:00 uur

door

Edoardo Costantini

geboren te Gallarate, Italië

Promotor:	prof. dr. K. Sijtsma	(Tilburg University)
Copromotores:	dr. K. M. Lang	(Utrecht University)
	dr. T. Reeskens	(Tilburg University)
Leden promotiecommissie:	prof. dr. A.G. de Waal	(Tilburg University)
	prof. dr. K. Van Deun	(Tilburg University)
	prof. dr. S. van Buuren	(Utrecht University)
	prof. dr. J.L. Ellis	(Open University)
	dr. J.R. van Ginkel	(Leiden University)

Table of Contents

Chapter 1

Introduction

This dissertation proposes methods for reducing the number of predictor variables in multiple imputation when scores from a large survey are missing. The methods use principal component analysis, thus aiming at preserving essential information from the larger predictor set. The proposed methods fit into the well-known multivariate imputation by chained equations (MICE) algorithm (van Buuren & Oudshoorn, 2000). In this chapter, we introduce the missing data problem in social science data, multiple imputation (MI), MICE, and the research objectives of the dissertation. Finally, the outline of the dissertation is provided.

1.1 Missing values and multiple imputation

Social scientists commonly collect data through surveys, which often suffer from nonresponse. When administering a social science survey, a group of respondents is asked several questions about their behaviors, opinions, and values. Respondents may fail or decide not to provide answers to certain questions—because they do not understand a question or refuse to disclose information, among other reasons (pp. 208–210, Groves et al., 2009)—which creates missing values in the data. When data have missing values even the most fundamental analyses cannot be performed.

Consider a researcher who wants to compute the correlation between the first two variables in the example data set in Figure 1.1. The researcher would not be able to estimate this correlation without first addressing the missing values. An intuitive solution, known as complete-case analysis, would be to discard the rows of the data with missing values on the first two variables and compute the correlation based on the sub-sample of respondents answering both questions. Even though this is a highly popular strategy among researchers (e.g., van Ginkel et al., 2010), it is wasteful. In this example, only four out of ten respondents have observed values for the first two variables and the more variables are involved in an analysis, the smaller the remaining sample will be. Using complete-case analysis to estimate a model involving all variables in the data presented in Figure 1.1 would result in a sample size of zero.

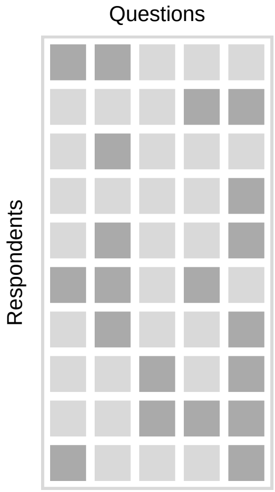


Figure 1.1: Representation of the shape of data social scientists usually analyze. Every square represents the answer given to a question by a respondent. The light gray squares represent observed values, while the dark gray squares represent missing values. Every row represents the record of the answers given by a respondent to all the questions. Every column represents the record of the answers given by the respondents to a question.

Another problem with the complete-case analysis is that it can bias estimation. Analyzing only the complete cases assumes that the respondents who answer all questions are equivalent to the respondents who do not. The correlation between the first two variables could be systematically higher (or lower) in the group of people who responded to both questions compared to the group of people who did not. In such a situation, analyzing the complete cases would result in a positively or negatively biased estimate of the correlation value in the population of interest.

Multiple imputation (MI; Rubin, 1976) is a powerful alternative to complete-case analysis to address the missing values in a data set. MI replaces the missing values in the original data with plausible values, multiple times. This way, multiple versions of the original data are defined. Figure 1.2 shows three imaginary imputed versions of the data in Figure 1.1. After obtaining these multiple imputations, the researcher can estimate the correlation between the first two variables on each of the multiple versions of the imputed data, this time using all respondents. Then, the multiple estimates of the correlation results are combined according to known *pooling* rules (Rubin, 1987) to obtain a single estimate of the correlation and its standard

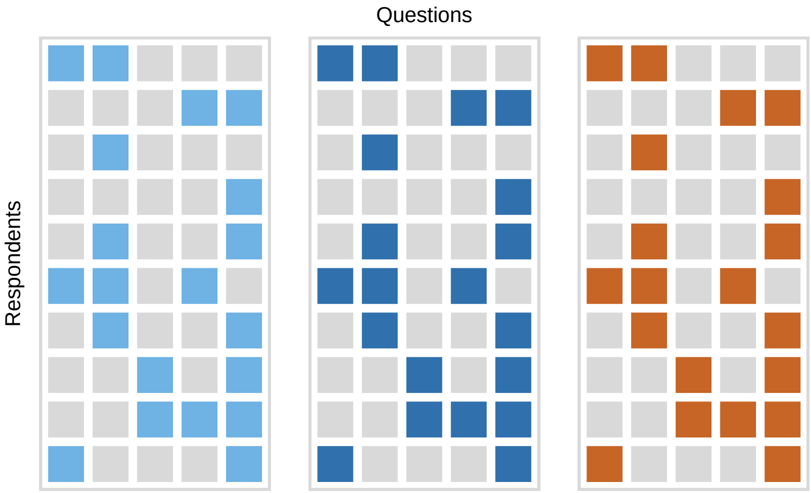


Figure 1.2: Differently imputed versions of the original data where the missing values have been replaced by different plausible values, or imputation.

error. The multiple imputations enable the pooled standard error to incorporate both the conventional sampling variance and the additional variance that is due to the missing data. Furthermore, by enabling the use of all available data together with the imputed values for the missing scores, MI results in higher estimation precision and power for hypothesis testing compared to complete-case analysis. Finally, if the right assumptions are met, MI can correct for the bias that would be introduced by using complete-case analysis.

In this dissertation, we focused on a specific implementation of MI known as full conditional specification (Raghunathan et al., 2001) or MICE (van Buuren & Oudshoorn, 2000). MICE is particularly suited to address problems of missing values in social science surveys because it can produce valid imputations for data sets containing variables of different measurement levels (van Buuren, 2007). This flexibility is rooted in how MICE obtains imputations for multivariate missing data through a collection of univariate regression models, rather than having to specify a joint model for all variables with missing values. These univariate regression models are known as imputation models and they define a separate univariate conditional distribution for every variable that needs to be imputed in a data set. Each imputation model can be defined as any prediction model that reflects the distribution of the variable under imputation. With MICE, a normally distributed variable can be imputed with linear regression, while a binary variable in the same data set can be imputed using logistic regression.

MICE requires researchers to make many decisions, most of which involve a degree of subjectivity. For example, MICE, like any other MI approach, relies on an assumption known as missing-at-random (MAR) which cannot be tested. Researchers must decide whether they believe the MAR assumption holds on a case-by-case basis and they can at most conduct sensitivity analyses to assess the impact a potential violations would have on the imputation results (Glynn et al., 2000; Heckman, 1976, 1979; N. Little, 2011; R. J. A. Little, 1993). Another subjective decision is whether the MICE algorithm has reached a state of convergence. MICE is an iterative algorithm that replaces the initial guesses for the missing values with increasingly more plausible guesses at every new iteration. After an unknown number of iterations, the new guesses start to fluctuate around a mean imputation, at which point the algorithm is usually considered to have converged. However, there is no numerical definition of this convergence and researchers have to decide whether convergence is attained based on visual inspection of graphical tools.

Apart from the characteristics of the data, the tenability of the MAR assumption and convergence hinge on the specification of the MICE algorithm. In particular, one critical specification affecting both the MAR assumption and convergence is the selection of predictors for the imputation models. Leaving important predictors out of the imputation models can result in a violation of the MAR assumption while including too many predictors can result in estimation problems (e.g., Hardt et al., 2012; White et al., 2011). As the number of variables available in a data set with missing values increases, the number of possible predictors can become too large. Multicollinearity may occur, and the estimation of the imputation models becomes unstable. This forces researchers to make difficult decisions about which predictors to include. Specialized software provides some data-driven approaches to help define the imputation model predictors, but even with the help of these tools, the selection of imputation model predictors remains one of the most challenging and delicate steps of the imputation procedure.

Despite its importance, the choice of the predictors in the imputation models is often neglected in many research articles using MI and published in high-profile social science journals. Only a few articles describe which variables are used as predictors in the imputation models (Mustillo, 2012; Mustillo & Kwon, 2015), and when they do, the exclusive use of the variables involved in the analysis seems to be the predominant choice (Costantini, Lang, Reeskens, & Sijtsma, 2023). To facilitate the specification of more adequate imputation models, social scientists need new tools.

1.2 High-dimensional prediction models

The problem of selecting the predictors for the imputation model can be cast as a high-dimensional prediction problem. High dimensionality occurs when the number of variables in a data set is much larger than the number of respondents. In this context, prediction models are afflicted by increased variance in parameter estimates, and risk of overfitting, among other estimation problems (e.g., Hastie et al., 2009, p. 46–47). These challenges can be addressed by various methodologies, including but not limited to variable selection (Dempster et al., 1977), regularized regression (Hoerl & Kennard, 1970; Tibshirani, 1996; Zou & Hastie, 2005), and the substitution of predictors with summary variables (Jolliffe, 2002, pp. 167–198).

In survey research, the number of respondents is often much larger than the number of questions asked, but the high-dimensional prediction literature can be used to establish more data-driven routines to define the predictors for the imputation models in MICE when imputing a survey with hundreds of questions. At the start of this research project, different uses of high-dimensional prediction had been proposed in the imputation literature but there was a lack of comparative studies showing which approach should be preferred in which context. Through the chapters in this dissertation, we delve into how high-dimensional modeling tools can be used to develop data-driven, easy-to-use routines to build imputation models for social and behavioral data.

1.3 Outline

In Chapter 2, we explored various approaches to automatically select the predictors for the imputation models when dealing with missing values in social science data. Our investigation included techniques such as subset selection, regularized regression, decision trees, and dimensionality reduction as strategies for constructing imputation models. To evaluate their performance, we conducted a Monte Carlo simulation considering bias and confidence interval coverage for estimates of common statistics in social science. Additionally, we utilized a resampling study to assess how these methods fared on data generated by an unknown data-generating model. Our results showed that regularization and subset selection where effective at identifying valuable predictors for the imputation model. However, principal components

(PCs) derived from fully observed variables and used as auxiliary predictors in the imputation models emerged as a more promising approach.

As proposed by Howard et al. (2015), the auxiliary-variables PCs approach tested in Chapter 2 uses PCA to exclusively process the fully observed potential auxiliary variables. This can be limiting as in social surveys fully observed variables are rare. An alternative use of PCA was implemented by the *PcAux* (Lang et al., 2018) R package to create a few summary predictors based on all variables in the data, not just the fully observed ones. The PcAux approach relies on a single imputation pre-processing step that temporarily fills in missing values to allow for the computation of the PCs based on all variables in the data. These PCs can then be used as predictors in the imputation models specified for MICE to generate multiple imputations for the original data. Yet another approach consists of computing PCs for every variable under imputation, at every iteration of the algorithm. Much like the PcAux method, this variable-by-variable approach enables the computation of PCs on all variables within the data set, not just the fully observed ones. This is achieved by computing PCs on the imputed versions of the variables at each iteration. In contrast to PcAux, this approach eliminates the need for pre-processing through single imputation. However, it comes with a higher computational load, as PCA is carried out for every variable in each iteration of the MICE algorithm.

In Chapter 3, we compared these different ways of using PCA to define the predictors for the imputation models of a MICE algorithm. We also investigated how their performance depends on the number of PCs used. Howard et al. (2015) presented imputation results using a single PC explaining approximately 40% of the auxiliary variables variance, but we expected that this heuristic rule would not apply to complex real-data applications. Through a simulation study, we examined which use of PCA in the MICE algorithm and which number of retained PCs resulted in better estimates of a selection of sufficient statistics for common analysis models in social science. The use of PCA for every variable at every iteration resulted in low bias and good confidence interval coverage, but its performance relied on using at least as many PCs as there were latent variables in the data-generating model.

In Chapter 4, we explored how the addition of supervision could improve upon the classical unsupervised PCA. In this context, supervision means using the variables that are under imputation to help compute PCs that are good predictors. A known issue of using PCA for prediction is that the most important PCs might capture variation that is unrelated to the outcome variable (Bair et al., 2006). This becomes a problem when the number of possible predictors is large but only a minor fraction of them contains useful information for predicting the outcome variable. Supervision helps by prioritizing PCs that predict well the dependent variable as well as summarize systematic variation in the set of possible predictors. We designed a simulation study to compare different approaches to supervised dimensionality reduction as imputation models for MICE. In particular, we focused on supervised principal component regression (SPCR, Bair et al., 2006), principal covariates regression (de Jong & Kiers, 1992), and partial least squares (Wold, 1975). We found that all forms of supervision obtained better imputations using fewer PCs than the original unsupervised PCA-based approach. We also found that among the different versions of supervised dimensionality reduction, SPCR resulted in lower bias and was closer to nominal confidence interval coverage when estimating a range of sufficient statistics for common analysis models in the social sciences.

In Chapter 5, we described an example imputation and analysis of the data collected by the European Values Study (EVS) using SPCR as an imputation model in MICE. For this chapter, we implemented an algorithm to estimate SPCR on data with variables of any measurement level (Costantini, 2023b) and we developed univariate imputation methods based on SPCR as experimental functions in a fork of the *mice* R package. We imputed the EVS data with the SPCR-based MICE and with a specification of MICE using carefully selected predictors for the imputation models. We compared the convergence trends of the two imputation procedures, the distribution of the imputed values, and the estimates of the parameters in a regression model estimated on the imputed data. The use of SPCR as a univariate imputation method resulted in similar point estimates to the approach with carefully selected predictors while offering a simpler imputation model specification, but it required much more computation time.

Each chapter of this dissertation is accompanied by an interactive results dashboard which provides an opportunity to explore the comprehensive results of each study. Readers are encouraged to engage with these dashboards, which can be installed as R packages following the instructions outlined at the onset of each chapter. More information regarding the availability and use of the software can be found in the references provided in the chapters.

Chapter 2

High-dimensional imputation for the social sciences: a comparison of state-of-the-art methods

Abstract Including a large number of predictors in the imputation model underlying a multiple imputation (MI) procedure is one of the most challenging tasks imputers face. A variety of high-dimensional MI techniques can help, but there has been limited research on their relative performance. In this study, we investigated a wide range of extant high-dimensional MI techniques that can handle a large number of predictors in the imputation models and general missing data patterns. We assessed the relative performance of seven high-dimensional MI methods with a Monte Carlo simulation study and a resampling study based on real survey data. The performance of the methods was defined by the degree to which they facilitate unbiased and confidence-valid estimates of the parameters of complete data analysis models. We found that using lasso penalty or forward selection to select the predictors used in the MI model and using principal component analysis to reduce the dimensionality of auxiliary data produce the best results.

Results dashboard To run the results dashboard accompanying this chapter install the Shiny app as an R package from the Zenodo permanent repository:

```r
# Install shiny app
devtools::install_url(
        "https://zenodo.org/records/10964103/files/EdoardoCostantini/plotmihd-v2.3.zip"
)
```

Then, you can start the app by running this command:

```r
# Start the app
plotmihd::start_app()
```

Reproducibility statement The R code used to produce the results reported in this chapter can be found in the permanent repository stored on Zenodo (Costantini, 2023c). The README.md file contains instructions on how to reproduce the results.

Costantini, E., Lang, K. M., Reeskens, T., & Sijtsma, K. (2023). High-dimensional imputation for the social sciences: A comparison of state-of-the-art methods. *Sociological Methods & Research*. https://doi.org/10.1177/00491241231200194

2.1 Introduction

Today's social, behavioral, and medical scientists have access to large multidimensional data sets that can be used to investigate the complex roles that social, psychological, and biological factors play in shaping individual and societal outcomes. Large social scientific data sets—such as the World Values Survey and the European Values Study (EVS)—are easily accessible to researchers, but making use of the full potential of these data requires dealing with the crucial problem of multivariate missing data.

2.1.1 The state of imputation in Sociology

Sociologists working with social surveys are usually interested in drawing inferential conclusions based on a substantively interesting analysis model. Generally, these analysis models require complete data, so the researcher must address any missing values before moving on to their substantive analysis. There are many possible missing data treatments from which to choose, and their relative strengths and weaknesses are covered elsewhere (e.g., Enders, 2010; R. J. A. Little & Rubin, 2002; van Buuren, 2018). In this paper, we will focus on Rubin's (1987) multiple imputation (MI), which is one of the most effective ways of addressing missing values in survey data.

MI is a three-step procedure that entails imputation, analysis, and pooling phases. The fundamental idea of the *imputation phase* is to replace each missing data point with d plausible values sampled from the posterior predictive distribution of the missing data, given the observed data. This phase generates d completed versions of the original data set that are each analyzed separately during the *analysis phase*, using any standard complete data analysis model. Finally, in the *pooling phase*, the d sets of estimates from the analysis models are pooled following Rubin's rules (Rubin, 1987) to create a single set MI parameter estimates and standard errors.

Missing values are one of the main factors impacting the quality of data gathered with surveys (Meyer et al., 2015), and nonresponse rates in large social survey have risen drastically over the last two decades (Brick & Williams, 2013; Massey & Tourangeau, 2013; Williams & Brick, 2018). To explore how sociologists are addressing the issue of nonresponse in their research, we reviewed how missing data have been discussed in the articles published over the last five years in two leading sociological journals: American Journal of Sociology (AJS) and American Sociological Review (ASR). We found that of the 148 AJS research articles that mentioned using a survey, or some form of sample, for inferential analysis, 24 addressed the presence of missing values, and 17 conducted some form of imputation. Of these 17, only 13 performed MI, and among these 13, only three articles gave information on which predictors were used in the imputation models. Turning to ASR, the picture was similar. Of the 191 research articles published between January 2017 and January 2022 that met the inclusion criteria described above, 20 reported performing MI. Of these 20 articles, only six gave information regarding which predictors were used in the imputation models. Across the two journals, in the nine papers we found that described which predictors were used in the imputation models, the predominant choice was to use only the analysis model variables in the imputation model.

In general, it seems that even when sociologists pay attention to the problem of missing values, little attention is given to which variables should be used in the imputation models. Similar conclusions were drawn in other literature reviews (Mustillo, 2012; Mustillo & Kwon, 2015). However, which variables to include in the imputation models is a crucial decision in MI. Leaving out important predictors of missingness can induce *missing not at random* (MNAR) data (Collins et al., 2001), while including good predictors can both correct for nonresponse bias and improve the efficiency of the parameters estimates (Collins et al., 2001; von Hippel & Lynch, 2013).

2.1.2 The challenge of specifying good imputation models

Specifying the imputation model is one of the most challenging steps in dealing with missing values. As described by van Buuren et al. (1999), the task involves defining two aspects of the model: the model form (e.g., linear, logistic) and the predictor matrix (i.e., the set of predictors that enter the imputation model). The first choice is straightforward in virtually any imputation task, as it depends primarily on the measurement level of the variables under imputation. The second choice requires a careful selection process aimed at identifying the subset of variables that will be most useful in a given imputation model.

Generally, the variables that will be part of the analysis model should also be included in the imputation model. When some analyzed variables (including transformations such as polynomials or interactions) are excluded from the imputation, the analysis and imputation models are said to be uncongenial (Meng, 1994). Such uncongeniality can lead to biased parameter estimates and invalid inferences. When designing an imputation model, the range of analysis models for which the resulting imputations will be congenial is an important consideration. In the methodological literature, this concept is known as the scope of the imputation model. van Buuren (2018, p. 46) distinguishes three typical imputation model scopes:

- *Narrow Scope:* Narrowly scoped imputation models are matched to individual analyses. In such a scenario, the imputation is a customized pre-processing step intended to facilitate only a single analysis model. When imputing with a narrow scope, the primary objective is to ensure that all the variables in the analysis models (including relevant transformations) appear in the imputation model. An analyst who imputes their own data and plans to estimate only one model (or a single series of nested models) may wish to specify a narrow scoped imputation model.
- *Intermediate Scope:* Imputation models with an intermediate scope are designed to support several different analysis models. The imputer will generally know approximately which analyses are intended but may not have an exhaustive list of all variables that will be analyzed. The objective is to design an imputation model that will be congenial with all planned and unplanned analysis models. Such analytic contexts frequently arise within research teams wherein several different analyses contribute to a larger research program. The evaluation of the Dating Matters intervention (Tharp, 2012) is an example of one such research program. Due to the size and complexity of the data and the diversity of the intended analyses, treating the missing data in the Dating Matters evaluation took several months of dedicated work (Niolon et al., 2019). The resulting imputations were then used to support the substantive analyses by which various dimensions of the intervention were evaluated (e.g., Estefan et al., 2021; Vivolo-Kantor et al., 2021).
- *Broad Scope:* Imputation models with a broad scope are designed to create imputations that will be congenial to the most general set of analysis models feasible. The imputer cannot know beforehand which variables will be part of the analysis models, so the imputation models are designed to be general enough to accommodate a wide range of potential analyses. Practically speaking, the objective is to recreate the moments of the hypothetically fully observed data as closely as possible. Rubin (1987, p. 3) originally envisioned MI as a method using broadly scoped imputation models to treat publicly released data and argues that well-implemented MI can accommodate models that were not contemplated by the imputer (R. J. A. Little & Rubin, 2002, p. 218). Any data

> curation institution imputing data that are intended for public release will need imputation models with a broad scope. The Federal Reserve Board's Survey of Consumer Finances (Kennickell, 2017) and the Luxembourg Wealth Study (LWS, 2020) are two examples of surveys released after performing MI, and used by sociologists publishing in AJS and ASR.

Despite its importance, congeniality should not be considered the sole guiding principle when defining imputation models. There are even cases where uncongeniality can improve on the efficiency of the standard complete data procedure, a phenomenon known as superefficiency (Meng, 1994, pp. 544–546; Rubin, 1996, p. 481; R. J. A. Little and Rubin, 2002, pp. 217–218). Furthermore, an imputation model that is congenial to a given analysis model may nevertheless fail to produce proper imputations. Rubin (1976, pp. 584–585) described the three conditions under which the distribution of the missingness is ignorable. The first of these conditions is that the missing data are missing at random (MAR), meaning that the probability of being missing is the same within groups defined by the observed data (i.e., conditioning on the observed data). When this condition is violated, standard MI can lead to biased parameter estimates, even if the analysis and imputation models are congenial.

Meeting the MAR assumption requires specifying imputation models that include the variables that correlate with the missingness and the analysis model variables. Omitting such variables from the imputation model results in imputation under MNAR (Collins et al., 2001, p. 339). Applying standard MI under MNAR can lead to bias in the parameter estimates and can invalidate inferences involving the imputed variables (Collins et al., 2001, pp. 341–343). Therefore, including as many good predictors of the variables under imputation as possible in the imputation model is generally advisable. In this study, we focus on methods that assume MAR data. However, a considerable amount of research has been devoted to developing missing data treatments for MNAR data. We refer interested readers to Enders's (2010) review of the two classes of MNAR models (i.e., selection models and pattern mixture models) and to N. Little's (2011) subsample ignorable multiple imputations: a method to obtain valid inferences with MI under MNAR under certain additional assumptions.

We refer to all variables that are not targets of imputation, as potential auxiliary variables. This set of potential auxiliaries may include important predictors of missingness, variables that correlate with the imputation targets, and variables that are not useful for imputation. Discerning which of the potential auxiliary variables may be useful predictors in the imputation model can be a daunting task. Following an inclusive approach (i.e., including numerous auxiliary variables in the imputation model) reduces the chances of omitting important correlates of missingness, thereby making the MAR assumption more plausible (Rubin et al., 1995, pp. 826–827; Schafer, 1997, p. 23; White et al., 2011; van Buuren, 2018, p. 167). Furthermore, Collins et al. (2001) showed that the inclusive strategy reduces estimation bias and increases efficiency. When designing broad and intermediate imputation models, the inclusive strategy can also grant congeniality with a wider range of analysis models.

Although following the inclusive strategy may be beneficial for the imputation procedure, it is often infeasible to use all potential auxiliaries as predictors with standard imputation methods. Standard imputation methods, such as imputation under the normal linear model (van Buuren, 2018, p. 68), face computational limitations in the presence of many predictors. For example, using traditional (unpenalized) regression models for the imputation model requires the number of predictors (p) in the imputation models to be smaller than the number of observed cases (n) to avoid mathematical singularity of the underlying system of equations (James et al., 2013, p. 203). As a result, imputers need to balance the benefits of the inclusive strategy with its computational limits. The large number of variables available in modern social scientific data sets makes the difficult step of deciding which predictors to include in the imputation models even more arduous.

In addition to their size, other aspects of social surveys and other social scientific data can further complicate the task of specifying good imputation models. Sociologists and researchers working with large social surveys often want to estimate analysis models that use composite scores (i.e., aggregates of multi-item scales). When working with multi-item scales, the imputer needs to decide if variables should be imputed at the item level or at the scale level. When all a scale's items are usually missing or observed together, scale-level imputation can be effective (Mainzer et al., 2021). When item-level missing predominates, however, the literature generally suggests imputing multi-item scales at the item level (Eekhout et al., 2014; Gottschall et al., 2012; van Buuren, 2010), but pursuing such a strategy can lead to increased dimensionality of the imputation models (Eekhout et al., 2018).

Furthermore, social surveys are often longitudinal, and it is usually most convenient to impute such a data structure in wide format (van Buuren, 2018, p. 312). A wide data set has a single record for each unit, with observations made at subsequent time points coded as additional columns in the data set. As a result, long-running panel studies might easily induce large pools of potential auxiliary variables with which the imputer must contend.

2.1.3 High-dimensional imputation

The factors discussed above—or combinations thereof—may result in *high-dimensional* imputation problems wherein the pool of potential auxiliary variables is larger than the available sample size. Such high-dimensional problems preclude a straightforward application of MI and force researchers to choose which variables to include in the imputation model or otherwise regularize the imputation model. One possible solution to this problem is using high-dimensional prediction models as the imputation model. When we say “high-dimensional prediction”, we are referring to the branch of statistical prediction concerned with improving prediction in situations where the number of predictors is larger than the number of observed cases (the so-called $p > n$ problem). Recent developments in high-dimensional imputation techniques leverage high-dimensional prediction methodology to offer opportunities for embracing an inclusive strategy while substantially diminishing its downsides.

MI has been combined with high-dimensional prediction models in algorithms that use shrinkage methods (Deng et al., 2016; Zhao & Long, 2016) and dimensionality reduction (Howard et al., 2015; Song & Belin, 2004) to avoid the obstacles of an inclusive strategy. Tree-based imputation strategies (Burgette & Reiter, 2010; Doove et al., 2014) also have the potential to overcome the computational limitations of the inclusive strategy. The nonparametric nature of decision trees bypasses the identification issues most parametric methods face in high-dimensional contexts. To the best of our knowledge, no study to date has directly compared the performance of the various high-dimensional MI methods recommended in the literature.

2.1.4 Scope of the current project

The goal of this project was to compare how different high-dimensional MI (HD-MI) methods fare when imputing data sets with many variables. In particular, we were interested in the types of imputation problems that may arise in large social scientific data sets. Such data sets do not need to be strictly high-dimensional to be *too large* for standard MI routines. Even in low-dimensional settings (i.e., $n > p$), including too many auxiliary variables in the imputation model can bias analysis model estimates and lead to convergence problems and other computational issues (Hardt et al., 2012). The high-dimensional imputation approaches we compared in this project can be used to simplify the process of specifying a good imputation model in both high- and low-dimensional problems.

We compared seven state-of-the-art HD-MI algorithms in terms of their ability to support statistically valid analyses. We chose these techniques because they stood out as the most promising candidates in our review of the HD-MI literature. The comparison was based on two numerical experiments: a Monte Carlo simulation study and a resampling study using Wave 5 of EVS. The simulation study allows us to compare the imputation methods in an artificial scenario with maximum experimental control. In a simulation study, we are able to precisely manipulate data features to match our experimental goals because we define the population model. However, the variables in a simulation study are usually sampled from simple multivariate distributions with regular, unrealistic mean and covariance structures. The resampling study allows us to shed the artifice of the simulation study and compare the methods using real social scientific data. EVS is a large-scale, cross-national survey on human values administered in approximately 40 countries across Europe. The EVS data contain both numerical and categorical variables associated via a complicated, heterogeneous covariance structure. Performing a resampling study on this data set allows us to estimate bias and coverage in a more ecologically valid—albeit still somewhat artificial—scenario than is possible with a Monte Carlo simulation study.[1]

The imputation techniques we compared are best suited to data-driven imputation with an intermediate or broad scope. The potential benefits of HD-MI methods lie in the automatic imputation model specification that these techniques offer. Therefore, we focused on data-driven imputation tasks where the objective is accommodating a wide range of analysis models. However, the techniques we compared do not exclude the possibility of specifying more narrowly scoped imputation models. With little tweaking, one can always force specific variables into the imputation model.

In what follows, we first introduce the missing data treatments that we compared in our study. Then, we present the methodology and results of the two numerical experiments, we discuss the implications of the results for applied researchers, and we provide recommendations. We conclude by discussing the limitations of the study and suggesting future research directions.

2.2 Imputation methods and algorithms

We use the following notation: scalars, vectors, and matrices are denoted by italic lowercase, bold lowercase, and bold uppercase letters, respectively. A scalar be-

---

[1] We chose to use the EVS data for a resampling study rather than an applied example because there is no ground truth in an applied example. The resampling study offers much of the ecological validity of an applied example with the added benefit of supporting the same types of generalizable conclusions provided by a simulation study.

longing to an interval is indicated by $s_1 \in [s_2, s_3]$, while a scalar taking the values in a set is represented as $s_1 \in \{s_2, s_3\}$. We use the scope resolution operator, ::, to designate a function provided by a specific software package. So, for example, *mice::quickpred()* represents the *quickpred()* function provided by the *mice* package.

Consider an $n \times p$ data set, $\mathbf{Z}$, comprising variables $\mathbf{z}_1$, $\mathbf{z}_2$, $\ldots$, $\mathbf{z}_p$. Assume that the first t variables of $\mathbf{Z}$ have missing values and that these t variables are the targets of imputation. Denote the columns of $\mathbf{Z}$ containing $\mathbf{z}_1$ to $\mathbf{z}_t$ as the $n \times t$ matrix, $\mathbf{T}$. The remaining $(p-t)$ columns of $\mathbf{Z}$ contain variables that are not targets of imputation. These variables constitute a pool of *potential* auxiliary variables that could be used to improve the imputation procedure. Let $\mathbf{A}$ be a $n \times (p-t)$ matrix denoting this set of potential auxiliary variables and write $\mathbf{Z}$ as $\mathbf{Z} = (\mathbf{T}, \mathbf{A})$. For a given $\mathbf{z}_j$, with $j = (1, \ldots, p)$, denote its observed and missing components as $\mathbf{z}_{j,\text{obs}}$ and $\mathbf{z}_{j,\text{mis}}$, respectively. Let $\mathbf{Z}_{-j} = (\mathbf{z}_1, \ldots, \mathbf{z}_{j-1}, \mathbf{z}_{j+1}, \ldots, \mathbf{z}_p)$ be the collection of $p-1$ variables in $\mathbf{Z}$ excluding $\mathbf{z}_j$. Denote by $\mathbf{Z}_{-j,\text{obs}}$ and $\mathbf{Z}_{-j,\text{mis}}$ the components of $\mathbf{Z}_{-j}$ corresponding to the data units in $\mathbf{z}_{j,\text{obs}}$ and $\mathbf{z}_{j,\text{mis}}$, respectively.

2.2.1 Multiple imputation by chained equations

Assume that $\mathbf{Z}$ is the result of n random samples from a multivariate distribution defined by an unknown set of parameters θ. The multiple imputation by chained equations (MICE) approach obtains the posterior distribution of θ by sampling iteratively from conditional distributions of the form $P(\mathbf{z}_1 | \mathbf{Z}_{-1}, \theta_1), \ldots, P(\mathbf{z}_t | \mathbf{Z}_{-t}, \theta_t)$, where $\theta_1, \ldots, \theta_t$ are imputation model parameters specific to the conditional distributions of each variable with missing values.

More precisely, the MICE algorithm takes the form of a Gibbs sampler[2]. At the mth iteration $(m = 1, \ldots, M)$, samples are drawn for the jth target variable $(j = 1, \ldots, t)$ from the following distributions:

$$\hat{\theta}_j^{(m)} \sim p(\theta_j | \mathbf{z}_{j,\text{obs}}, \dot{\mathbf{Z}}_{-j,\text{obs}}^{(m)}) \tag{2.1}$$

$$\mathbf{z}_{j,\text{mis}}^{(m)} \sim p(\mathbf{z}_{j,\text{mis}} | \dot{\mathbf{Z}}_{-j,\text{mis}}^{(m)}, \hat{\theta}_j^{(m)}), \tag{2.2}$$

where $\hat{\theta}_j^{(m)}$ and $\mathbf{z}_{j,\text{mis}}^{(m)}$ are draws from the parameter's full conditional posterior distribution (2.1) and the missing data posterior predictive distribution (2.2), respectively. $\dot{\mathbf{Z}}_{-j,\text{obs}}^{(m)}$ and $\dot{\mathbf{Z}}_{-j,\text{mis}}^{(m)}$ are subsets of the variables in $\mathbf{Z}_{-j}^{(m)}$ (potentially every variable

---

[2] Technically, the MICE algorithm is only a true Gibbs sampler when there exists a valid joint distribution for all targets of imputation. When such a joint distribution does not exist, the MICE algorithm is still valid, but it is not a true Gibbs sampler.

in $\mathbf{Z}_{-j}^{(m)}$). These subsets are chosen by the imputer to act as predictors in the elementary imputation model for $\mathbf{z}_j$. After convergence, d sets of values are sampled from (2.2) and used as imputations. Any analysis model can then be estimated on each of the d completed data sets, and the parameter estimates can be pooled using Rubin's rules (Rubin, 1987).

In the following, we describe all the missing data treatments we compared in this study. First, we describe the seven *high-dimensional MICE strategies* we compared in this study. They follow the general MICE framework, but they differ in which elementary imputation methods they use to define Equations (2.1) and (2.2). Second, we describe three *benchmark mice strategies*, which are well-established approaches in the field of sociology and the missing data treatment literature. Finally, we describe two *benchmark non-MI strategies*, which are important baselines of comparisons that do not rely on imputation.

2.2.2 High-dimensional MICE strategies

2.2.2.1 MICE with step-forward selection

A linear regression model is the standard univariate imputation model for MICE. However, ordinary linear regression (OLS) faces computational limitations when applied to data sets with many predictors. If n is not much larger than p, the regression estimates will have large variances, and, if $p > n$, there is no unique solution for the regression coefficients. Researchers have been studying model-building strategies to overcome these limitations for decades (e.g., Dempster et al., 1977). One of these strategies, known as forward stepwise subset selection (Efroymson, 1966), has been implemented in the popular imputation software IVEware (Raghunathan et al., 2002). We refer to this method as MI Step-Forward (MI-SF).

Forward selection identifies the subset of the predictors that are most related to the dependent variables by iteratively evaluating the improvement in fit contributed by including each additional predictor. Starting with an empty imputation model, MI-SF iteratively adds the variable that most increases the model-explained variance. New predictors are added as long as the additional proportion of variance they explain exceeds a specified threshold value R^2_{min}. As a result, MI-SF ensures that the predictors included in Equation (2.1) must explain some non-trivial proportion variability in the variable under imputation. The value of R^2_{min} used in the MI-SF algorithm is fixed across iterations, but the imputation model for every variable might change between iterations.

2.2.2.2 MICE with a fixed ridge penalty

So-called *shrinkage methods* represent an alternative to subset selection (see Hastie et al., 2009, pp. 62–79 for a review). These methods address the computational problems caused by large numbers of predictors by shrinking the estimated coefficients toward zero. Ridge regression (Hoerl & Kennard, 1970) is a common shrinkage method that imposes a penalty during model estimation to shrink the regression slopes toward zero and allow a large number of predictors to be included in the model, while still controlling the variance of the estimates. When applied to the imputation model in MICE, a ridge penalty allows a more inclusive auxiliary variable strategy.

MICE with a fixed ridge penalty uses the Bayesian normal linear model described by van Buuren (2018, p. 68, algorithm 3.1) as the univariate imputation method. We refer to this method as Bayesian Ridge (BRidge). In this approach, the sampling of each $\hat{\theta}_j^{(m)}$ in Equation (2.1) relies on inverting the cross-products matrix of $\dot{\mathbf{Z}}_{-j,\text{obs}}^{(m)}$[3]. Adding a positive constant (the ridge penalty, κ) to the diagonal of the cross-product matrix stabilizes this inversion. Indeed, if $p > n$, sufficiently large values of κ will facilitate inversion of the cross-products matrix and induce a unique (albeit biased) solution for the regression coefficients.

In BRidge, every variable in $\mathbf{Z}_{-j}$ is used as a predictor in the imputation model, and the ridge penalty is the only precaution taken to address a large number of predictors. The value of κ is usually chosen to be close to 0 (e.g., $\kappa = 0.0001$), because values larger than 0.1 may introduce excessive systematic bias (van Buuren, 2018, p. 68). However, larger values of κ may be necessary to adequately stabilize the estimation in certain scenarios. In the present work, we chose the value of κ by means of cross-validation.

2.2.2.3 Direct use of regularized regression

Lasso regression (*least absolute shrinkage and selection operator*; Tibshirani, 1996) is another popular shrinkage method. Unlike ridge regression, the lasso penalty achieves both shrinkage and automatic variable selection (whereas ridge does not exclude any variables). The extent of the lasso penalization depends on a tuning parameter, λ, which is selected from a set of possible values by means of cross-validation. For sufficiently large values of λ, lasso will force some coefficient estimates to be exactly zero thereby excluding the associated predictors from the

---

[3] When estimating ridge regression coefficients, predictors are centered and scaled to have unit variance

fitted model. When applied to an imputation model, lasso will automatically select which predictors enter the imputation model. Zhao and Long (2016) and Deng et al. (2016) used lasso regression as the univariate imputation model in a MICE algorithm to impute high-dimensional data and referred to this approach as *direct use of regularized regression* (DURR[4]).

At iteration m, for a target variable $\mathbf{z}_j$, DURR replaces Equations (2.1) and (2.2) with the following two steps:

1. Generate a bootstrap sample $\mathbf{Z}^{*(m)}$ by sampling with replacement from $\mathbf{Z}^{(m)}$, and train a regularized linear regression model (such as lasso regression) with $\mathbf{z}^{*(m)}_{j,\text{obs}}$ as outcome and $\mathbf{Z}^{*(m)}_{-j,\text{obs}}$ as predictors[5]. This produces a set of parameter estimates (regression coefficients and error variance), $\hat{\theta}^{(m)}_j$, that can be viewed as a sample from Equation (2.1).
2. Use $\mathbf{Z}^{(m)}_{-j,\text{mis}}$ and $\hat{\theta}^{(m)}_j$ to predict $z_{j,\text{mis}}$, and obtain draws from the posterior predictive distribution of the missing data as in equation (2.2).

Hence, at every iteration, each elementary imputation model is estimated as a lasso regression, and uncertainty regarding the parameter values is included by bootstrapping.

In high-dimensional cases, lasso selects at most n predictors (Zou & Hastie, 2005). So, when using lasso for imputation, no elementary imputation model will contain more predictors than the number of observed cases on the corresponding outcome. Deng et al. (2016) compared lasso with the elastic net—which does not have this restriction—for high-dimensional MI, but they did not find evidence to favor the elastic net over lasso. Lasso is also computationally simpler than the elastic net because lasso only has one tuning parameter to estimate whereas the elastic net has two. Therefore, we chose to implement DURR with lasso as the regularization method.

2.2.2.4 Indirect use of regularized regression

While DURR simultaneously performs model regularization and parameter estimation in Equation (2.1), the *indirect use of regularized regression* (IURR[6]; Deng et al.,

---

[4] The DURR approach is now implemented in the R package *mice* via the *mice::mice.impute.lasso.norm()* and *mice::mice.impute.lasso.logreg()* elementary imputation methods.

[5] As with ridge regression, predictors are centered and scaled to have unit variance.

[6] The IURR approach is now implemented in the R package *mice* via the *mice::mice.impute.lasso.select.norm()* and *mice::mice.impute.lasso.select.logreg()* elementary impu-

2016; Zhao & Long, 2016) algorithm uses regularized regression exclusively for variable selection. The selected variables are then used as predictors in the imputation models of a standard MI procedure.

At iteration m, the IURR algorithm performs the following steps for each target variable, $\mathbf{z}_j$:

1. Fit a linear regression model using a regularized method that does variable selection (e.g., lasso). Take $\mathbf{z}_{j,\text{obs}}$ as the dependent variable and $\mathbf{Z}^{(m)}_{-j,\text{obs}}$ as the predictors (unlike DURR, IURR uses the original data, not a bootstrap sample). The regression coefficients that are *not* shrunk to 0 define the active set of variables that will be used as predictors in the actual imputation model (i.e., the variables in $\dot{\mathbf{Z}}^{(m)}_{-j}$).

2. Obtain the maximum likelihood estimates of the regression coefficients and the error variance from the linear regression of $\mathbf{z}_{j,\text{obs}}$ onto the active set of predictors defined in step 1. Then, sample new values of these parameters from a multivariate normal distribution parameterized by the MLEs[7]:

   $$(\hat{\theta}^{(m)}_j, \hat{\sigma}^{(m)}_j) \sim N(\hat{\theta}^{(m)}_{\text{MLE}}, \hat{\mathbf{\Sigma}}^{(m)}_{\text{MLE}}) \tag{2.3}$$

   so that Equation (2.3) corresponds to Equation (2.1) in the general MICE framework.

3. Impute $\mathbf{z}_{j,\text{mis}}$ by sampling from the posterior predictive distribution based on $\dot{\mathbf{Z}}^{(m)}_{-j,\text{mis}}$ and the parameters' posterior draws, $(\hat{\theta}^{(m)}_j, \hat{\sigma}^{(m)}_j)$.

DURR uses regularized regression to directly obtain $\hat{\theta}^{(m)}_j$, a procedure that inherently induces estimation bias. Compared to DURR, IURR separates the variable selection step, which involves using the biasing penalty term, from the sampling of the imputation model parameters. Assuming the variable selection step does not exclude any important predictors, the two-step approach of IURR could outperform DURR by using unbiased estimates of $\hat{\theta}^{(m)}_{\text{MLE}}$ and $\hat{\mathbf{\Sigma}}^{(m)}_{\text{MLE}}$ to define the posterior distributions of the imputation model parameters. IURR effectively establishes a data-driven decision rule to select imputation model predictors while avoiding the direct involvement of the biasing penalty in the simulation of a random draw from Equation (2.1).

---

tation methods.

[7] The sampling notation is the same used by Deng et al. (2016).

2.2.2.5 MICE with Bayesian lasso

Zhao and Long (2016) proposed the MICE with Bayesian lasso imputation algorithm (BLasso), an MI procedure that uses the Bayesian lasso as its elementary imputation method: MICE with Bayesian lasso (BLasso). A Bayesian lasso model is a regular Bayesian multiple regression model with informative priors on the slope coefficients that allow interpreting the mode of the slopes' posterior distribution as lasso estimates (Hans, 2009; T. Park & Casella, 2008). Following Zhao and Long (2016), we used the Bayesian lasso specification given by Hans (2010b). Given data with a sample size, n, a dependent variable, $\mathbf{y}$, and a set of predictors, $\mathbf{X}$, the Bayesian lasso model has the following form.

$$p(\mathbf{y}|\beta, \sigma^2, \tau) = \mathrm{N}(\mathbf{y}|\mathbf{X}\beta, \sigma^2\mathbf{I}_n) \tag{2.4}$$

$$p(\beta_j|\tau, \sigma^2, \rho) = (1-\rho)\delta_0(\beta_j) + \rho\left(\frac{\tau}{2\sigma}\right) \times \exp\left(\frac{-\tau\left\|\beta_j\right\|_1}{\sigma}\right) \tag{2.5}$$

$$\sigma^2 \sim \text{Inverse-Gamma}(a, b) \tag{2.6}$$

$$\tau \sim \text{Gamma}(r, s) \tag{2.7}$$

$$\rho \sim \text{Beta}(g, h) \tag{2.8}$$

Equation (2.4) represents the density function of a multivariate normal random variable with mean $\mathbf{X}\beta$ and covariance matrix $\sigma^2\mathbf{I}_n$ evaluated at $\mathbf{y}$. Equation (2.5) is the mixture prior distribution for the regression coefficients β_j proposed by Hans (2010b). This formulation differs from the classical Bayesian lasso prior proposed by T. Park and Casella (2008) because of the presence of the sparsity parameter, ρ (Ley and Steel, 2009, pp. 655–656; Scott and Berger, 2010, pp. 2592), and the point mass at zero, $\delta_0(\beta_j)$. Finally, Equations (2.6) to (2.8) represent hyper priors for the residual variance, σ^2, the penalty parameter, τ, and the sparsity parameter, ρ. Our implementation of BLasso imputation replaced Equation (2.1) with the BLasso model defined by Equations (2.4) to (2.8) with $\mathbf{y} = \mathbf{z}_{j,\text{obs}}$ and $\mathbf{X} = \mathbf{Z}_{-j,\text{obs}}$.

The R code used to perform the BLasso imputation was based on the R Package *blasso* (Hans, 2010a) and can be found in the code repository for this article (Costantini, 2023c). For a detailed description of the Bayesian lasso MI algorithm in a univariate missing data context see Zhao and Long (2016).

2.2.2.6 MICE with PCA

By extracting principal components (PCs) from the set of potential auxiliary variables, $\mathbf{A}$, the *MICE with PCA* (MI-PCA) method summarizes the information contained in $\mathbf{A}$

with just a few components. These PCs can then be used as predictors in a standard, low-dimensional application of MICE. The MI-PCA procedure can be summarized as follows:

1. Extract the first PCs that cumulatively explain the desired proportion of the variance in the set of potential auxiliary variables, $\mathbf{A}$[8], and collect these components in a new matrix, $\mathbf{A}'$;
2. Replace $\mathbf{A}$ in $\mathbf{Z}$ with $\mathbf{A}'$ to obtain $\mathbf{Z}' = (\mathbf{T}, \mathbf{A}')$;
3. Use the standard MICE algorithm with a Bayesian normal linear model and no ridge penalty to obtain multiply imputed data sets from $\mathbf{Z}'$.

The MI-PCA method was inspired by Howard et al. (2015) and the *PcAux* R package (Lang et al., 2018). For this study, we used the R function *stats::prcomp()* to perform the PCA estimation via truncated singular value decomposition. Hence, $p > n$ data are not a problem. When $\mathbf{A}$ has more columns than rows, *prcomp()* will simply extract a maximum of n components.

2.2.2.7 MICE with classification and regression trees

MICE with classification and regression trees (MI-CART; Burgette & Reiter, 2010) is a MICE algorithm that uses classification and regression trees (CART) as the elementary imputation method. Given an outcome variable $\mathbf{y}$ and a set of predictors $\mathbf{X}$, CART is a nonparametric recursive partitioning technique that models the relationship between $\mathbf{y}$ and $\mathbf{X}$ by sequentially splitting observations into subsets of units with relatively more homogeneous $\mathbf{y}$ values. At every splitting stage, the CART algorithm searches through all variables in $\mathbf{X}$ to find the best binary partitioning rule to predict $\mathbf{y}$. The resulting collection of binary splits can be visually represented by a decision tree structure where each terminal node (or *leaf*) represents the conditional distribution of $\mathbf{y}$ for units that satisfy the splitting rules.

For each $\mathbf{z}_j$, the mth iteration of MI-CART proceeds as follows:

1. Train a CART model to predict $\mathbf{z}_{j,\text{obs}}$ from the corresponding $\mathbf{Z}^{(m)}_{-j,\text{obs}}$.
2. Assign each element of $\mathbf{z}_{j,\text{mis}}$ to a terminal node by applying the splitting rules from the fitted CART model to $\mathbf{Z}^{(m)}_{-j,\text{mis}}$.

---

[8] The columns of $\mathbf{A}$ are centered and scaled to have unit variance.

3. Create imputations for each element of $\mathbf{z}_{j,\text{mis}}$ by sampling from the pool of $\mathbf{z}_{j,\text{obs}}$ in the terminal node containing $\mathbf{z}_{j,\text{mis}}$. This procedure corresponds to sampling from the missing data posterior predictive distribution in Equation (2.2).

This approach does not consider uncertainty in the imputation model parameters since the tree structure is not perturbed between iterations. Therefore, MI-CART cannot produce proper imputations in the sense of Rubin (1986). The implementation of MI-CART used in this paper corresponds to the one presented by Doove et al. (2014, p. 95, algorithm 1) and the *impute.mice.cart()* function from the *mice* package.

CART searches for the best splitting criterion one variable at a time. As a result, $p > n$ does not pose the same computational limitations that plague methods based on linear regression. More variables can increase estimation times but will not result in computational obstructions.

2.2.2.8 MICE with random forests

MICE with random forests (MI-RF) is a MICE algorithm that uses random forests as the elementary imputation method. The random forest algorithm (e.g., Hastie et al., 2009, p. 588) entails fitting many decision trees (e.g., CART models) to subsamples of the original data. These subsamples are derived by resampling rows with replacement and sampling subsets of columns without replacement. The random forest algorithm results in an ensemble of fitted decision trees that generate a sample of predictions for each outcome value. Consequently, random forests often demonstrate better prediction performance than individual trees by reducing the variance of the estimated prediction function.

For each $\mathbf{z}_j$, the mth iteration of MI-RF proceeds as follows:

1. Generate k bootstrap samples from $\mathbf{Z}_{-j,\text{obs}}$.
2. Use these bootstrap samples to fit k single trees predicting $\mathbf{z}_{j,\text{obs}}$ from a random subset of the variables in $\mathbf{Z}_{-j,\text{obs}}$.
3. Generate a pool of k terminal nodes for each element of $\mathbf{z}_{j,\text{mis}}$ by applying the splitting rules from each of the k fitted trees to the appropriate columns of $\mathbf{Z}_{-j,\text{mis}}$.
4. Create imputations for each element of $\mathbf{z}_{j,\text{mis}}$ by sampling from the $\mathbf{z}_{j,\text{obs}}$ contained in the pool of terminal nodes defined above.

Bootstrapping and random input selection introduce uncertainty regarding the imputation model parameters (i.e., the tree structure), as required by a proper MI procedure. For more details on the MI-RF algorithm, see Doove et al. (2014, p. 103). To perform MICE with random forests we used the R function *mice::impute.mice.rf().* As with CART, the random forests algorithm is not subject to computational limitations in high-dimensional problems because random forests simply aggregate a collection of univariate decision trees.

2.2.3 Benchmark MICE strategies

2.2.3.1 MICE with quickpred

A simple way to select predictors for an imputation model is to include variables that relate to the nonresponse or explain a considerable amount of variance in the targets of imputation. One popular implementation of this idea is to select as predictors those variables whose association with the variables under imputation, or their response indicators, exceeds some threshold. This selection strategy was proposed by van Buuren et al. (1999) and has been implemented in the *quickpred* function provided by the popular R package *mice* (van Buuren, 2018, p. 267). We refer to this approach as MI-QP. As both an intuitive, pragmatic option and the default method of selecting predictors in one of the most popular MI software packages, MI-QP represents an important benchmark against which to compare the performance of the more theoretically sound approaches described above.

The MI-QP approach has two main drawbacks. First, selecting predictors based on their correlations with the targets of imputation and the associated response indicators can still select collinear, redundant predictors. If one predictor is highly correlated with another and with a variable under imputation, both will be selected. Second, when applied to $p > n$ scenarios, MI-QP is not guaranteed to select fewer predictors than observations available for a given imputation model. As a result, MI-QP often needs to be augmented by other techniques to address collinearity and linear dependencies in the data.

2.2.3.2 MICE with analysis model variables as predictors

According to our review of the articles published in AJS and ASR, a common approach to address the large number of possible predictors is to use only the analysis model variables in the imputation model. We refer to this approach as MI-AM. Consider a researcher working with EVS data who wants to estimate a linear model by regressing one item on 10 others afflicted by non-response. The MI-AM imputation strategy would imply using only these 11 variables in the imputation models, instead of manually searching all of the 250 variables contained in the survey for meaningful imputation predictors.

The MI-AM strategy ensures the congeniality of the analysis and imputation models. Furthermore, as long as the analysis model does not include more variables than the number of observed cases, MI-AM is not affected by the dimensionality of the data. However, by following this strategy, any MAR predictors that are not part of the analysis model will be excluded from the imputation. In such cases, the MAR assumption is violated, and the missingness is MNAR.

2.2.3.3 Oracle MICE

As hinted by the previous two approaches, the MI literature recommends following three principles to decide which predictors to include in the imputation models (van Buuren, 2018, p. 168):

1. Include all variables that are part of the analysis model(s).
2. Include all variables that are related to the nonresponse.
3. Include all variables that are correlated with the targets of imputation.

In practice, the first criterion can be met only if the analysis model is known before imputation, which is not always true. Furthermore, researchers can never be sure that the second criterion is entirely met, as there is no way to know exactly which variables are responsible for missingness. However, with simulated data, we know which variables define the response model. The oracle MICE approach (MI-OR) is an ideal specification of the MICE algorithm that uses this knowledge to include only the relevant predictors in the imputation models. As such, this method cannot be used in practice, but it provides a useful reference point for the desirable performance of an MI procedure. The MI-OR imputations were generated using the Bayesian normal linear model as the univariate imputation method.

2.2.4 Non-MI strategies

2.2.4.1 Complete-case analysis

By default, most data analysis software either fails in the presence of missing values or defaults to analyzing only the complete cases (R Core Team, 2020; The pandas development team, 2020). As the default behavior of most statistical software, complete-case analysis (CC) remains a popular missing data treatment in the social sciences (T. D. Little et al., 2013; Peugh & Enders, 2004). CC can also be a useful approach in certain scenarios (White & Carlin, 2010). For example, when the analysis model is a linear regression of y onto a set of predictors, $\mathbf{X}$, CC yields valid inferences if the missingness depends only on $\mathbf{X}$ and not on y (R. J. A. Little and Rubin, 2002, p. 43; N. Little, 2011). However, even in this case, CC can be inefficient as it uses a reduced sample size compared to what could be used through proper imputation (R. J. A. Little and Rubin, 2002, p. 42; Schafer and Graham, 2002). Furthermore, unless the data are missing completely at random (MCAR), CC can bias parameter estimates (Rubin, 1987, p. 8, Schafer and Graham, 2002). Nevertheless, the continued popularity of CC makes it an important benchmark method.

2.2.4.2 Gold standard

We also estimated the analysis models directly on the fully observed data before imposing any missing values. In the following, we refer to the results obtained in this fashion as the gold standard (GS). These results represent the counterfactual analysis that would have been performed if there had been no missing data.

2.3 Simulation study

We investigated the performance of the methods described above with a Monte Carlo simulation study. Following a similar procedure to that employed by Collins et al. (2001), we generated $S = 1000$ samples of $n = 200$ units while varying two experimental factors: the number of variables in the data set, $p \in \{50, 500\}$, and the proportion of missing cases on each of the incomplete variables, $pm \in \{0.1, 0.3\}$. Table 2.1 summarizes the four resulting crossed conditions.

We chose the values of n and p to reflect extreme dimensionality situations that would tease apart the relative strengths and weaknesses of the imputation methods considered here. Nonetheless, we selected these values to be somewhat plausible for real-world social scientific studies. Consider, for example, that a typical EVS wave has around 55,000 observations and 250 items in its questionnaire. Therefore, data structures similar to those in both our low- and high-dimensional conditions could arise by taking reasonable subsets of EVS data (potentially over several waves). As for the levels of pm, we chose the lower level to match the 10% of missing cases that is typical of variables in EVS data. We also included a more extreme level to create more challenging—but still realistic—conditions for the imputation methods.

condition	label	n	p	pm
1	low-dim-low-pm	200	50	0.1
2	high-dim-low-pm	200	500	0.1
3	low-dim-high-pm	200	50	0.3
4	high-dim-high-pm	200	500	0.3

Table 2.1: Summary of conditions for Experiment 1. Low-dim (high-dim) represent conditions where the number of predictors is smaller (larger) than the number of observations available. Low-pm (high-pm) represent conditions where the proportion of missing values is low (high)

For every iteration, we imposed missing values on six target items, and then we used all missing data treatment methods described above to obtain estimates of the means, variances, and covariances of these incomplete variables.

2.3.1 Simulation study procedure

2.3.1.1 Data generation

At every replication, a data matrix $\mathbf{Z}_{n \times p}$ was generated according to a multivariate normal model with means equal to 5 and unit variances. The distribution was centered around 5 as typical 10-point numerical items in the EVS data set have means around 5. After sampling the data, all variables were rescaled to have a variance of approximately 5, which reflects the typical size of the variance of 10-point items in the EVS data. For the correlation structure, we defined three blocks of variables based on three strengths of association: strong, weak, and none. The first five variables were strongly correlated ($\rho = 0.6$) among themselves; variables 6 to 10 were weakly correlated ($\rho = 0.3$) with the first 5 variables and among themselves; the remaining $p - 10$ variables were uncorrelated with any other variable in the data set. Of course, real survey data have more complex correlation structures than what we defined for this study. However, when specifying imputation models for survey data, the main challenge is often finding a few important auxiliary variables in a large collection of possible predictors. We defined the population correlation matrix with the three-block structure described above to replicate this type of situation in an experimentally unequivocal way.[9].

---

[9] We used fixed correlation levels instead of varying the correlation values (e.g., to manipulate collinearity) for the same reason. Varying the strengths of association within blocks would have diluted

2.3.1.2 Missing data imposition

Missing values were imposed on six of the items in $\mathbf{Z}$: three variables in the block of highly correlated variables $\{\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3\}$ and three in the block of lowly correlated variables $\{\mathbf{z}_6, \mathbf{z}_7, \mathbf{z}_8\}$. Item nonresponse was imposed by sampling from a Bernoulli distribution with individual probabilities of nonresponse defined by

$$p_{miss} = p(z_{i,j} = miss|\tilde{\mathbf{Z}}) = \frac{exp(\gamma_0 + \tilde{\mathbf{z}}_i\gamma)}{1 + exp(\gamma_0 + \tilde{\mathbf{z}}_i\gamma)}, \tag{2.9}$$

where $z_{i,j}$ is the ith subject's response on $\mathbf{z}_j$, $\tilde{\mathbf{z}}_i$ is a vector of responses to the set of missing data predictors for the ith individual, γ_0 is an intercept parameter, and γ is a vector of slope parameters. $\tilde{\mathbf{Z}}$ was specified to include two fully observed variables from the strongly correlated set and two from the weakly correlated set $\{\mathbf{z}_4, \mathbf{z}_5, \mathbf{z}_9, \mathbf{z}_{10}\}$. Therefore, the probability of nonresponse for a variable depended on variables present in the data, but never on the variable itself. As a result, when the elements of $\tilde{\mathbf{Z}}$ are included as predictors in the MI procedures, the MAR assumption is satisfied. All slopes in γ were fixed to 1, while the value of γ_0 was chosen through numerical optimization to produce the desired proportion of missing values[10].

2.3.1.3 Imputation

We generated ten imputed data sets by imputing the missing values with all methods described in the preceding section. To evaluate the convergence of the imputation models, we ran ten replications of the *high-dim-high-pm* condition and generated trace plots of the imputed values' means. The implementation of MI-SF in IVEware does not provide trace plots. Therefore, we plotted the distributions of the imputed values across 30 imputation chains against the observed data at iterations 1, 5, 10, 20, 40, 80, 160, 240, and 320. Based on the information provided by density and trace plots, we considered all of the imputation algorithms to have converged after 50 iterations.

IVEware does not offer any data-driven procedure for selecting R^2_{min}; and the IVEware authors recommend comparing results obtained with different R^2_{min} values. To optimize the performance of MI-SF, we tuned this parameter with a cross-validation procedure. We applied MI-SF with different R^2_{min} values (i.e., $10^{-1}, 10^{-2}, \ldots, 10^{-7}$),

---

this key feature of our population model.

[10] The pseudo R-squared for the logistic regression of the missing value indicator on the predictors of missingness was around 15%. The AUC for the logistic regression was around 0.72.

and we selected the value that resulted in the smallest average fraction of missing information (FMI; Rubin, 1987, eq. 3.1.10) across the analysis model parameters. The same cross-validation strategy was used to choose the value of the ridge penalty in the BRidge algorithm. We considered the values $10^{-1}, 10^{-2}, \ldots, 10^{-8}$ as candidates for the BRidge penalty parameter.

Both IURR and DURR could have been implemented with a variety of penalties (e.g., lasso, Tibshirani, 1996; elastic net, Zou and Hastie, 2005; adaptive lasso, Zou, 2006). In this study, we used lasso as it is computationally efficient, and it performed well for imputation in Zhao and Long (2016) and Deng et al. (2016). A 10-fold cross-validation procedure was used at every iteration of DURR and IURR to choose the penalty parameter. To maintain consistency with previous research, we specified the BLasso hyper-parameters in Equations (2.6), (2.7), and (2.8) as in Zhao and Long (2016): $(a, b) = (0.1, 0.1)$, $(r, s) = (0.01, 0.01)$, and $(g, h) = (1, 1)$. For the MI-PCA algorithm, the set of possible auxiliary variables in $\mathbf{A}$ was defined by all the fully observed variables. Another important decision when using PCA is the number of components to keep. Howard et al. (2015) used only the first component in their simulations. Since this component explained, on average, 40% of the variance in the auxiliary data, they recommend using enough components to explain 40% of the variance. For our study, we generated more complex data for which a single component was not likely to suffice. We, therefore, applied the intuitively appealing—albeit arbitrary—heuristic of using enough components to explain 50% of the total variance in the data.

Running MI-QP in the high-dimensional procedure led to frequent convergence failures. A more common use of the method includes accompanying the quickpred approach with a ridge penalty and data-driven checks that exclude collinear variables. We decided to run MI-QP in this more favorable manner by applying the *mice* package's usual data-screening procedures. Accordingly, the *mice()* call for MI-QP was specified with Bayesian normal linear regression as a univariate imputation method and with default values for the following arguments: ridge = $1e-5$, eps = $1e-4$, threshold = 0.999. Finally, we implemented the MI-AM method by applying the *mice::mice()* function to only the analysis model variables with Bayesian normal linear regression as a univariate imputation method. In this simulation study, the analysis model variables are the variables with missing values for which we wanted to estimate the means, variances, and covariances.

2.3.1.4 Analysis and comparison criteria

The *analysis model* comprised the joint distribution of the six variables with missing values. Therefore, we refer to these six incomplete variables as the *analysis model variables* below. After imputation, we estimated the six means, six variances, and 15 covariances for these variables on each imputed data set and pooled the estimates via the Rubin (1987) pooling rules. We then compared the performances of the imputation methods by computing the bias, confidence interval coverage, and confidence interval width for each estimated parameter.

Since we generated multivariate normal data, the sample means, variances, and covariances were the sufficient statistics for the joint distribution of the analysis model variables. Hence, we can infer that a method which demonstrates good performance when estimating these statistics will perform equally well when estimating other parameters that describe the same joint distribution. For example, the slopes, $\beta = \Sigma_X^{-1}\Sigma_{X,y}$, intercept, $\alpha = \mu_y - \mu_X^T\beta$, and residual variance, $\sigma_\varepsilon^2 = \sigma_y^2 - \beta^T\Sigma_X\beta$ of a general linear model can be defined directly in terms of these statistics. Using only this mean vector and covariance matrix, we could also factor analyze these six variables (Bartholomew et al., 2011, pp. 53–55) or estimate their structural relations via a structural equation model (Bollen, 1989, pp. 104–106). Importantly, the inverse implication does not generally hold. For example, in the special case noted above wherein CC can produce unbiased slope estimates, the estimated means, variances, and covariances of the underlying data could still be biased unless the data were MCAR. By focusing our analysis on a general set of sufficient statistics, we dissociated our results from any specific statistical model or test and increased the generalizability of our findings.

For a given parameter of interest θ, we used the absolute percent relative bias (PRB) to quantify the estimation bias introduced by the imputation procedure:

$$\textit{PRB} = \left| \frac{\bar{\hat{\theta}} - \theta}{\theta} \right| \times 100, \tag{2.10}$$

where θ is the true value of the focal parameter defined as $\sum_{s=1}^{S} \hat{\theta}_s^{GS}/S$, with $\hat{\theta}_s^{GS}$ being the Gold Standard parameter estimate for the sth repetition. The averaged focal parameter estimate under a given missing data treatment was computed as $\bar{\hat{\theta}} = \sum_{s=1}^{S} \hat{\theta}_s/S$, with $\hat{\theta}_s$ being the estimate obtained from the treated incomplete data in the sth repetition. Following Muthén et al. (1987), we considered PRB > 10 as indicative of problematic estimation bias.

To assess the performance in hypothesis testing and interval estimation, we evaluated the confidence interval coverage (CIC) of the true parameter value:

$$\mathit{CIC} = \frac{\sum_{s=1}^{S} I(\theta \in \widehat{\mathit{CI}}_s)}{S}, \tag{2.11}$$

where $\widehat{\mathit{CI}}_s$ is the confidence interval of the parameter estimate $\hat{\theta}_s$ in the sth repetition, and $I(.)$ is the indicator function that returns 1 if the argument is true and 0 otherwise.

CICs below 0.9 are usually considered problematic for 95% CIs (van Buuren, 2018, p. 52; Collins et al., 2001, p. 340) as they imply inflated Type I error rates. High CICs (e.g., above 0.99) indicate CIs that are too wide, implying inflated Type II error rates. Therefore, we considered CIs to show severe under-coverage (over-coverage) if CIC < 0.9 (CIC > 0.99). From a testing perspective, a CIC can be considered as significantly different from the nominal coverage rate if the magnitude of its difference from the nominal coverage proportion (p_0) is more than two times the standard error of p_0, $\mathit{SE}(p_0) = \sqrt{p_0(1-p_0)/S}$ (Burton et al., 2006). In our simulation study, the nominal coverage probability was 95%. Therefore, we considered 95% CI coverages outside the interval $[0.94, 0.96]$ to be significantly different from the nominal coverage rate. We assumed normal sampling distributions for variances and covariances when computing and pooling their CIs. This assumption is plausible under large sample conditions.

We also reported the average width of the confidence intervals (CIW), an indicator of statistical efficiency. An imputation method with a narrower confidence interval indicates higher efficiency and is therefore preferable. Nevertheless, the narrower CIW should not come at the expense of a lower than nominal CIC (van Buuren, 2018, p. 52).

2.3.2 Results

We computed both PRB and CIC for each of the 27 parameters in the analysis model (six means, six variances, and 15 covariances). To summarize the results, we focus on the expected and extreme values of these measures. In Figures 2.1 and 2.2, we report the average, minimum, and maximum PRB and CIC obtained with the different missing data treatments, for each parameter type. As the GS estimates were used to define the “true” values of the parameters, the bias for this method was by definition 0. So, we do not include bias of the GS estimates in the figure. For ease of presentation, we report the results only for the large proportion of missing cases ($pm = 0.3$) condition. While the relative performances were independent of the missing data rate, the performance patterns were clearer with a larger proportion of missing values. The interested reader can examine the same figures for the low proportion of missing cases ($pm = 0.1$) condition in the results dashboard.

2.3.2.1 Means

The largest PRB for the means was below 10 for all imputation methods. Only CC produced problematic degrees of bias. Looking at the relative performances, IURR, BRidge, MI-PCA, MI-SF, and MI-OR resulted in smaller biases than the other methods. In terms of CIC, only MI-PCA and MI-OR showed a consistently strong performance. Neither method demonstrated any extreme under-/over-coverage (i.e., all CICs $\in [0.9, 0.99]$), and both methods resulted in only the highest coverage being significantly different from nominal coverage (max CIC > 0.96).

IURR resulted in significant under-coverage of the true means (CICs < 0.94) in both the high-dimensional ($p = 500$) and low-dimensional ($p = 50$) conditions, although under-coverage was never severe (with CICs $\in [0.90, 0.94]$). MI-SF resulted in similarly trivial under-coverage and always returned CICs $\in [0.90, 0.94]$. DURR and BLasso demonstrated some significant differences from nominal coverage in the low-dimensional condition, and both led to extreme under-coverage in the high-dimensional condition. The tree-based methods and CC performed most poorly. These methods led to CICs significantly different from nominal coverage rates in all conditions, and they demonstrated extreme under-coverage even in the low-dimensional condition. MI-QP resulted in close to nominal coverage in the low-dimensional condition, performing about as well as MI-OR and MI-PCA. However, in virtually all replications of the high-dimensional condition, the CIs contained the true parameter values, thereby producing severe over-coverage. Finally, MI-AM resulted in significant to extreme under-coverage. This method was not influenced by the dimensionality of the data as it used the same six variables as predictors in both conditions.

2.3.2.2 Variances

IURR, BLasso, and the tree-based MI methods resulted in low biases (i.e., PRB < 10) in both the high and low dimensional conditions. For BLasso, these low biases were paired with low deviations from nominal coverage rates. IURR only demonstrated problematic CICs for the high-dimensional condition, where it produced extreme under-coverage (largest CIC < 0.9). MI-CART and MI-RF did not produce reasonable coverage in either the low- or the high-dimensional condition, with the largest coverage being significantly different from nominal (CICs < 0.94) and the smallest being severely below the nominal level (CICs < 0.9).

MI-PCA and MI-SF showed acceptable biases and reasonable coverage rates in the low-dimensional condition, but they showed large biases and under-coverage in the high-dimensional condition. In the low-dimensional condition, DURR produced low bias and reasonable coverage (i.e., only the lowest coverage being significantly different from nominal), but it resulted in PRBs > 10 for all variances in the high-dimensional condition, where it also produced extreme CI under-coverage. BRidge and CC performed poorly in nearly all conditions. These methods tended to demonstrate substantial biases and extreme under-coverage. Although MI-QP performed well in the low-dimensional condition, in the high-dimensional condition, it resulted in PRBs larger than 50 and CICs close to 1 for all six item variances. MI-AM maintained low bias and acceptable coverage for all item variances.

2.3.2.3 Covariances

MI-PCA was the only method that showed consistently strong performance when estimating covariances. MI-PCA showed negligible bias and minimal deviations from nominal coverage in both low- and high-dimensional conditions. In particular, the PRB was smaller than 10 for all covariances in both conditions and was almost as low as the PRB obtained by MI-OR. MI-PCA never produced extreme under-/over-coverage, and when the CIC was significantly different from the nominal rate, the CIs showed mild *over*-coverage (i.e., CICs greater than 0.96 but smaller than 0.99). After MI-PCA, IURR and MI-SF demonstrated the next strongest performance, with negligible bias and acceptable coverage in the low-dimensional condition. However, in the high-dimensional condition, IURR produced large biases and extreme under-coverage with the average bias being above the 10% threshold and even the largest coverage being just around the 90% threshold. In the high-dimensional condition, MI-SF showed a similar, albeit less severe, deterioration in performance.

Both MI-QP and BRidge displayed low bias and acceptable coverage in the low-dimensional condition, but they resulted in unacceptable biases in the high-dimensional condition. In the high-dimensional condition, MI-QP led to 100% coverage of the true values, while BRidge led to extreme under-coverage of the true values. MI-AM, DURR, BLasso, and the tree-based MI methods tended to result in PRBs larger than 10, accompanied by under-coverage of the true covariance values, even in the low-dimensional conditions.

2.3.2.4 Confidence interval width

In Figure 2.3, we report the CIW obtained with the different missing data treatments, averaged per parameter type across the repetitions. All methods maintained similar CIW independent of the dimensionality of the data. The two exceptions were MI-QP and Bridge. While the average CIW for MI-QP in the low-dimensional condition was in line with that of all the other methods, the CIW obtained with this method for all parameter types became larger than 10 for $p = 500$. In the high-dimensional case, the item variance CIWs obtained by Bridge were four times as large as those obtained in the low-dimensional scenario.

2.3.3 A note on collinearity

Following feedback provided by a reviewer of an earlier draft of this article, we included an additional simulation study to explore the effect of collinearity. We used the same simulation procedure described above, but we adjusted some of the design parameters. We fixed the proportion of missing cases to the highest value ($pm = 0.3$), as this factor did not affect the relative performances of the methods. We varied the number of columns in the data ($p \in \{50, 500\}$) and the strength of the correlation between the potential auxiliary variables ($\rho_{pav} \in \{0, 0.6, 0.8, 0.9\}$). Correlations higher than 0.6 are unlikely in survey data, but including the higher levels provides the opportunity to explore how the imputation methods perform when faced with problematic levels of collinearity.

In Figure 2.4, we report the average, minimum, and maximum PRB and CIC of the 15 covariances between two imputed items. In this report, we focus on the high-dimensional condition ($p = 500$), and we omit $\rho_{pav} = 0$, which can be considered equivalent to the results already reported. The interactive dashboard (Costantini, 2024c) contains the complete set of results. The relative performances of the methods were mostly unchanged. However, a few key differences should be noted. First, shrinkage-based methods resulted in lower PRB and closer-to-nominal CIC for higher levels of ρ_{pav}. In particular, the high PRB and low CIC that characterized BRidge in the original study results were mitigated as ρ_{pav} increased. For $\rho_{pav} = 0.9$, the highest bias returned by BRidge was lower than 10, and the lowest CIC was higher than 0.80. Similar trends arose for IURR and DURR, for which higher values of ρ_{pav} led to a lower PRB and closer-to-nominal CIC. Second, for higher values of ρ_{pav}, the PRB and CIC from MI-PCA essentially mirrored those of MI-AM. Finally, in the high-dimensional condition ($p = 500$), MI-QP had a prohibitively long imputation time. In a small trial run of the simulation, MI-QP required around 360 minutes

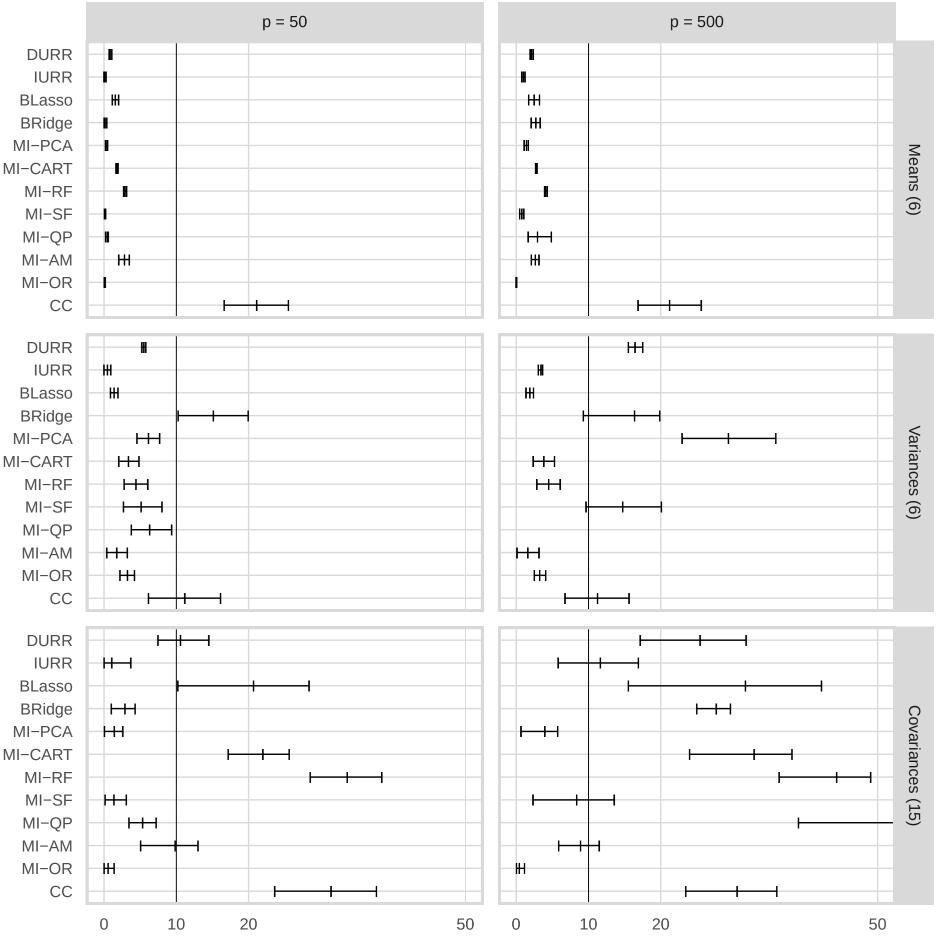


Figure 2.1: Minimum, average, and maximum absolute percent relative bias (PRB) for the six item means, six variances, and 15 covariances in the simulation study. If no data points are reported for a method in a panel, all of its PRBs were larger than 50. The methods reported on the Y-axis are: direct use of regularized regression (DURR), indirect use of regularized regression (IURR), MICE with Bayesian lasso (BLasso), MICE with Bayesian ridge (BRidge), MICE with principal component analysis (MI-PCA), MICE with CART (MI-CART), MICE with random forests (MI-RF), MICE with step-forward selection (MI-SF), oracle low-dimensional MICE (MI-OR), MICE with quickpred (MI-QP), MICE with analysis model (MI-AM), oracle low-dimensional MICE (MI-OR), and complete-case analysis (CC).

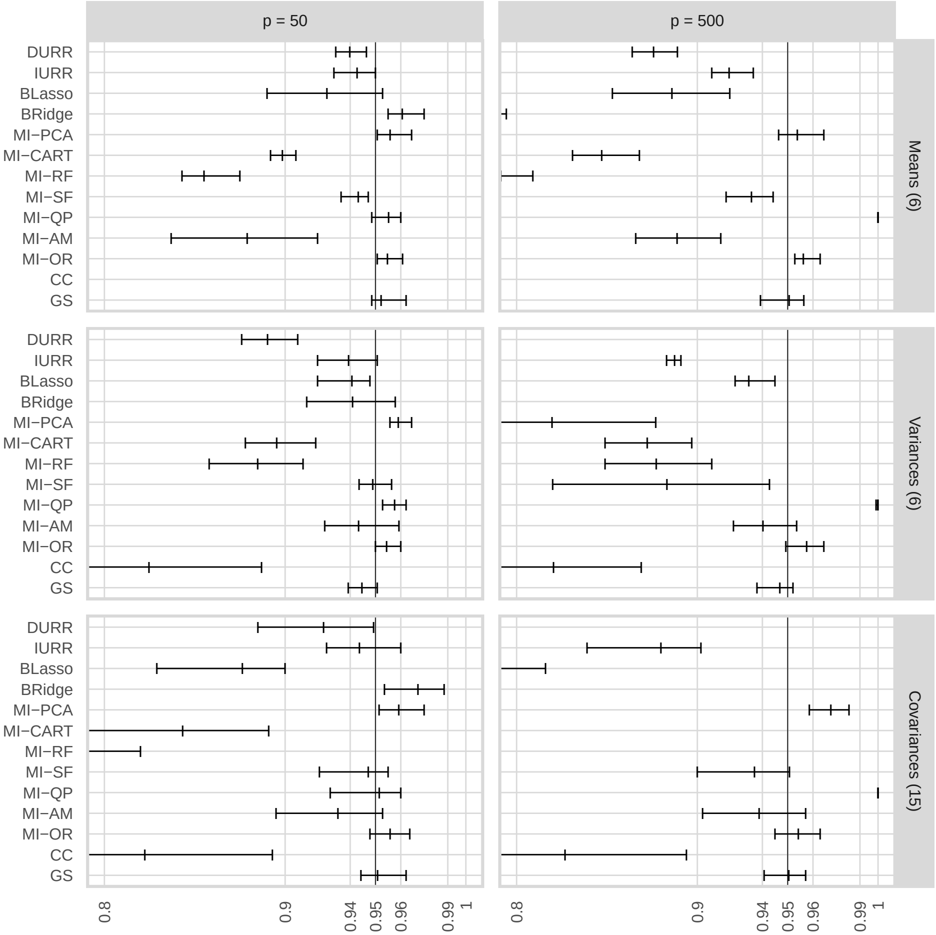


Figure 2.2: Minimum, average, and maximum confidence interval coverage (CIC) for the six item means, six variances, and 15 covariances in the simulation study. If no data points are reported for a method in a panel, all of its CICs were smaller than 0.80. The methods reported on the Y-axis are: direct use of regularized regression (DURR), indirect use of regularized regression (IURR), MICE with Bayesian lasso (BLasso), MICE with Bayesian ridge (BRidge), MICE with principal component analysis (MI-PCA), MICE with CART (MI-CART), MICE with random forests (MI-RF), oracle low-dimensional MICE (MI-OR), MICE with step-forward selection (MI-SF), MICE with quickpred (MI-QP), MICE with analysis model (MI-AM), oracle low-dimensional MICE (MI-OR), complete-case analysis (CC), and gold standard analysis (GS).

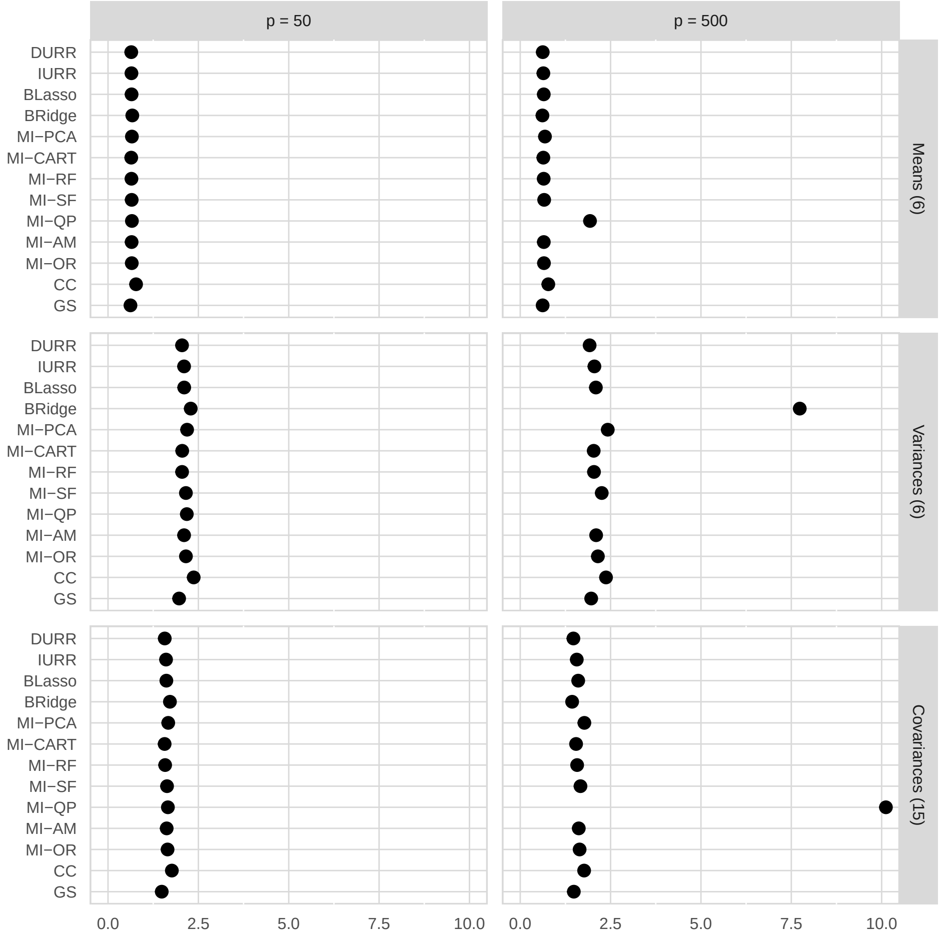


Figure 2.3: Average confidence interval width (CIW) across the six item means, six variances, and 15 covariances in the simulation study. If no data points are reported for a method in a panel, its CIW was larger than 10. The methods reported on the Y-axis are: direct use of regularized regression (DURR), indirect use of regularized regression (IURR), MICE with Bayesian lasso (BLasso), MICE with Bayesian ridge (BRidge), MICE with principal component analysis (MI-PCA), MICE with CART (MI-CART), MICE with random forests (MI-RF), MICE with step-forward selection (MI-SF), MICE with quickpred (MI-QP), MICE with analysis model (MI-AM), oracle low-dimensional MICE (MI-OR), and complete-case analysis (CC), and gold standard analysis (GS).

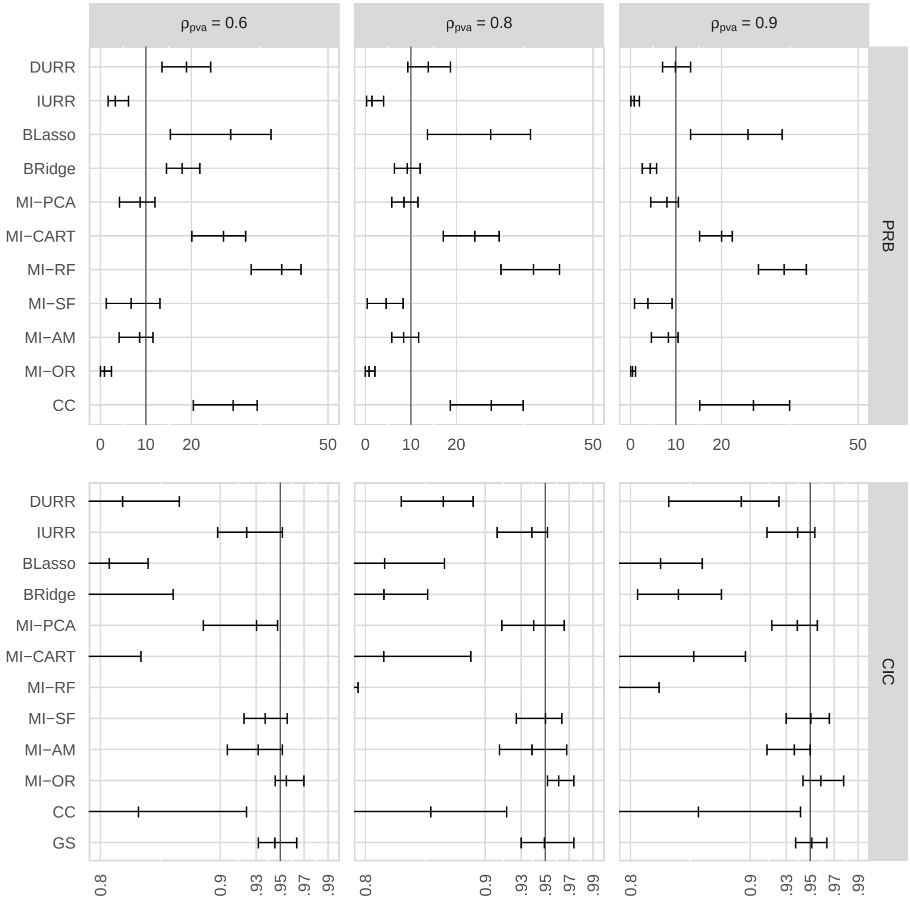


Figure 2.4: Minimum, average, and maximum absolute percent relative bias (PRB) and confidence interval coverage (CIC) across 15 covariances estimated between items with imputed values. The methods reported on the Y-axis are: direct use of regularized regression (DURR), indirect use of regularized regression (IURR), MICE with Bayesian lasso (BLasso), MICE with Bayesian Ridge (BRidge), MICE with principal component analysis (MI-PCA), MICE with CART (MI-CART), MICE with random forests (MI-RF), MICE with step-forward selection (MI-SF), MICE with analysis model (MI-AM), oracle low-dimensional MICE (MI-OR), complete-case analysis (CC), and gold standard analysis (GS).

to impute a single data set generated with $\rho_{pav} = 0.6$ and around 1130 minutes to impute a data set generated with $\rho_{pav} = 0.9$. IURR and MI-SF, the next two most computationally intensive methods, each took around 10 minutes to impute these data sets. Consequently, we included MI-QP only in the low dimensional condition of this additional simulation study.

2.4 EVS resampling study

We performed a resampling study based on the EVS data to assess whether the results of our simulation study would replicate in more realistic data. EVS is a high-quality survey widely used by sociologists for comparative studies between European countries (EVS, 2020b). Furthermore, it is freely available and represents the type of data social scientists regularly analyze. Variables in the EVS data are discrete numerical and categorical items following a variety of distributions.

To perform the resampling study, we treated the original EVS data as a population. We then resampled $S = 1000$ data sets of n units from this population, and we used these replicates as we used the multivariate normal samples in the simulation study. For each replicate, we imposed missing values, and we treated these missing values with the same methods explored in the simulation study. This procedure was repeated for low-dimensional and high-dimensional conditions. As the number of predictors in the data was fixed at $p = 243$, we controlled the dimensionality of the data by varying the sample size ($n \in \{1000, 300\}$). When the sample size was 300, after dummy coding categorical predictors, even a small proportion of missing values ($pm = 0.1$) led to a high-dimensional ($p > n$) situation. Although $n = 300$ might be too low to represent the typical use of EVS data, we do not see this as a limitation on the following results, for two reasons. First, our purpose in conducting this resampling study was primarily to see if our simulation results would carry over into data generated from a more realistic population model, not necessarily to see if those results would hold in a typical social science data set. Increasing the sample size to match ranges typically seen in analyzes of EVS data would remove the high-dimensional condition where we saw the most interesting results in the simulation, thereby greatly reducing the utility of the resampling study. Second, many social science studies analyze data with around 300 observations, so our samples are not unrealistic in a general sense.

2.4.1 Resampling study procedure

2.4.1.1 Data preparation and sampling

We used the third prerelease of the 2017 wave of EVS data (EVS, 2020a) to create a population data set with *no missing values*. The original data set contained 55,000 observations from 34 countries. We selected only the four founding countries of the

European Union included in the data set (France, Germany, Italy, and the Netherlands) because keeping all countries would have entailed either including a set of 33 dummy codes in the imputation models or imputing under some form of a multilevel model. Since both of these options fall outside the scope of the current study, we opted to subset the data as described. We excluded all columns that contained duplicated information (e.g., recoded versions of other variables), or metadata (e.g., time of the interview, mode of data collection).

The original EVS data set contained missing values. We needed to treat these missing data before we could use the EVS data in the resampling study. We used the *mice* package to fill the missing values with a single round of predictive mean matching (PMM). We used the *quickpred* function, to select the predictors for the imputation models. We implemented the variable selection by setting the minimum correlation threshold in *quickpred* to 0.3. The number of iterations in the *mice()* run was set to 200. We used a single imputation, and not MI, because this imputation procedure was used only to obtain a set of pseudo-fully observed data to act as the population in our resampling study and not for statistical modeling, estimation, or inference with respect to the true population from which the EVS data were sampled. For the same reason, the relatively poor performance that we observed for MI-QP in the simulation study is not relevant here. At the end of the data cleaning process, we obtained a pseudo-fully observed data set of 8,045 observations across four countries with $p = 243$ variables. For every replicate in the resampling study, we generated a bootstrap sample by sampling n observations with replacement from this data set.

2.4.1.2 Analysis models

To define plausible analysis models, we reviewed the models reported in the repository of publications using EVS data that is available on the EVS website (EVS, 2020b). As a result, we defined two linear regression models. Model 1 was inspired by Köneke (2014). The dependent variable was a 10-point item measuring euthanasia acceptance (‘Can [euthanasia] always be justified, never be justified, or something in between?’). The predictors included an item measuring the self-reported importance of religion in one’s life, trust in the health care system, trust in the state, trust in the press, country, sex, age, education, and religious denomination. A researcher might estimate this model to test a hypothesis regarding the effect of religiosity on the acceptance of end-of-life treatments.

Model 2 was inspired by Immerzeel et al. (2015). The dependent variable was a harmonized variable that quantifies the respondents' tendencies to vote for left- or right-wing parties, expressed on a 10-point left-to-right continuum. The predictors included a scale measuring respondents' attitudes toward immigrants and immigration ('nativist attitudes scale'). The scale was obtained by taking the average of respondents' agreement, on a scale from 1 to 10, with three statements: 'immigrants take jobs away from natives', 'immigrants increase crime problems', and 'immigrants are a strain on welfare system'. The remaining predictors were: attitudes toward law and order, attitudes toward authoritarianism, interest in politics, level of political activity, country, sex, age, education, employment status, socioeconomic status, importance of religion in life, religious denomination, and the size of the town where the interview was conducted. A researcher might estimate this model to test a hypothesis regarding the effect of xenophobia on voting tendencies.

2.4.1.3 Missing data imposition

We imposed missing data on six variables using the same strategy as in the simulation study. The targets of missing data imposition were the two dependent variables in models 1 and 2 (i.e., euthanasia acceptance, and left-to-right voting tendency), religiosity, and the three items making up the "nativist attitudes" scale. The response model was the same as in Equation (2.9), and three variables were included in $\tilde{\mathbf{Z}}$: age, education, and an item measuring trust in new people[11]. We chose these predictors because older people tend to have higher item nonresponse rates than younger people, and lower educated people tend to have higher item non-response rates than higher educated people (de Leeuw et al., 2003; Guadagnoli & Cleary, 1992). We also assumed that people with less trust in strangers would have a higher nonresponse tendency as they are likely to withhold more information from the interviewer (a stranger).

2.4.1.4 Imputation

We treated the missing values with the same methods used in the simulation study. MI-AM used all the variables present in either of the analysis models as predictors for the imputation models. MI-PCA was performed considering all the fully observed variables as possible auxiliary variables. In other words, the six variables with missing values were used in their raw form, while the remaining 237 were used to extract PCs. The other imputation methods were parameterized in the same way as in the

---

[11] The pseudo R-squared for the logistic regression of the missing value indicator on the predictors of missingness was around 14%. The AUC for the logistic regression was around 0.75.

simulation study, and convergence checks were performed in the same way. These convergence checks suggested that the imputation models had converged after 60 iterations.

2.4.2 Results

When estimating linear regression models, all partial regression coefficients can be influenced by missing values on a subset of the variables included in the model. Therefore, it is important to evaluate the estimation bias and CIC rates for all model parameters. Figure 2.5 reports the absolute PRBs for the intercept and all partial slopes from Model 2 obtained after using each imputation method, for both the low- and high-dimensional conditions. Model 2 has an intercept and 13 regression coefficients. Every horizontal line in the figure represents the PRB for the estimation of one of these 14 parameters. Figures 2.6 and 2.7 report CIC and CIW results in the same way. For ease of presentation, results for Model 1 are reported only in the results dashboard.

As seen in Figure 2.5, in both the high- and low-dimensional conditions, DURR, IURR, BLasso, MI-CART, and MI-SF showed only slightly larger PRBs than MI-OR. However, even MI-OR did not provide entirely unbiased parameter estimates. After imputing with MI-OR, almost half of the parameters in Model 2 were estimated with large bias (PRB > 10%). MI-PCA, MI-RF, and CC showed similar trends but produced larger PRBs (particularly CC). BRidge demonstrated the same results described in the simulation studies. It was competitive in the low-dimensional scenario, but it was inadequate with high-dimensional data (all PRBs > 10). In the low-dimensional condition, MI-QP resulted in only three parameter estimates with acceptable bias and only one in the high-dimensional condition. MI-AM resulted in six parameter estimates with acceptable bias in the low-dimensional condition but only one in the high-dimensional condition.

As shown in Figure 2.6, MI-SF, MI-OR, and DURR resulted in the lowest deviations from nominal coverage, with only one or two coverages differing significantly from the nominal level. IURR showed a similar trend but four coverages were significantly different from nominal in the low-dimensional condition.

BLasso, MI-PCA, MI-CART, MI-RF, MI-SF, and MI-AM all showed similar performance in the low-dimensional condition. These methods all significantly over-covered most of the parameters but did not produce any extreme under-/over-coverage, except for one parameter for MI-RF. BLasso, MI-PCA, and MI-RF maintained similar performance in the high-dimensional condition, but MI-CART improved to match the performance of MI-OR, and MI-AM produced extreme over-coverage for most of the parameters. BRidge performed well in the low-dimensional condition—around the level of IURR—but produced very poor coverages in the high-dimensional condition. MI-QP performed poorly in both the low- and high-dimensional conditions, producing only two non-significant coverages in the low-dimensional condition and none in the high-dimensional condition. CC performed quite well, but it had a much more pronounced tendency toward *under*-coverage than the MI methods. Notably, very few of the CICs fell into the range of extreme under-/over-coverage. Only the high-dimensional estimates from BRidge and MI-AM consistently exhibited extreme under-/over-coverage.

Finally, the average CIW for every parameter estimate is reported in Figure 2.7. In the low-dimensional condition, all methods result in similar CIWs. All methods result in larger confidence intervals in the high-dimensional condition reflecting a natural loss of information due to the smaller sample size used. However, Bridge, MI-QP, and MI-AM show drastically larger CIWs for the majority of the parameters.

2.4.2.1 Imputation time

Figure 2.8 reports the average imputation time for the different methods. IURR and DURR were the most time-consuming methods, with imputation times above one hour in the low-dimensional condition. In the high-dimensional condition, IURR and DURR were not as time-intensive due to the smaller sample size but still took more than ten times longer than MI-PCA and BLasso. MI-PCA was the fastest method, with imputation times of under a minute in both the high- and low-dimensional conditions. BLasso, MI-OR, and MI-AM were close seconds, with imputation times of two minutes or less in both conditions. BRidge, MI-CART, MI-RF, MI-SF, and MI-QP fell in the middle, with imputations times ranging from 3.5 (MI-CART) to 15.8 (MI-SF) minutes in the low-dimensional condition and from 1.2 (MI-CART) to 12.8 (MI-QP) minutes in the high-dimensional condition.

2.5 Discussion

2.5.1 Methods that work well

On balance, IURR, MI-SF, and MI-PCA were the strongest performers across the simulation study and the resampling study. In the simulation study, IURR and MI-SF produced trivial estimation bias for all parameters in the low-dimensional condition and for the means in the high-dimensional condition. Furthermore, the covariance

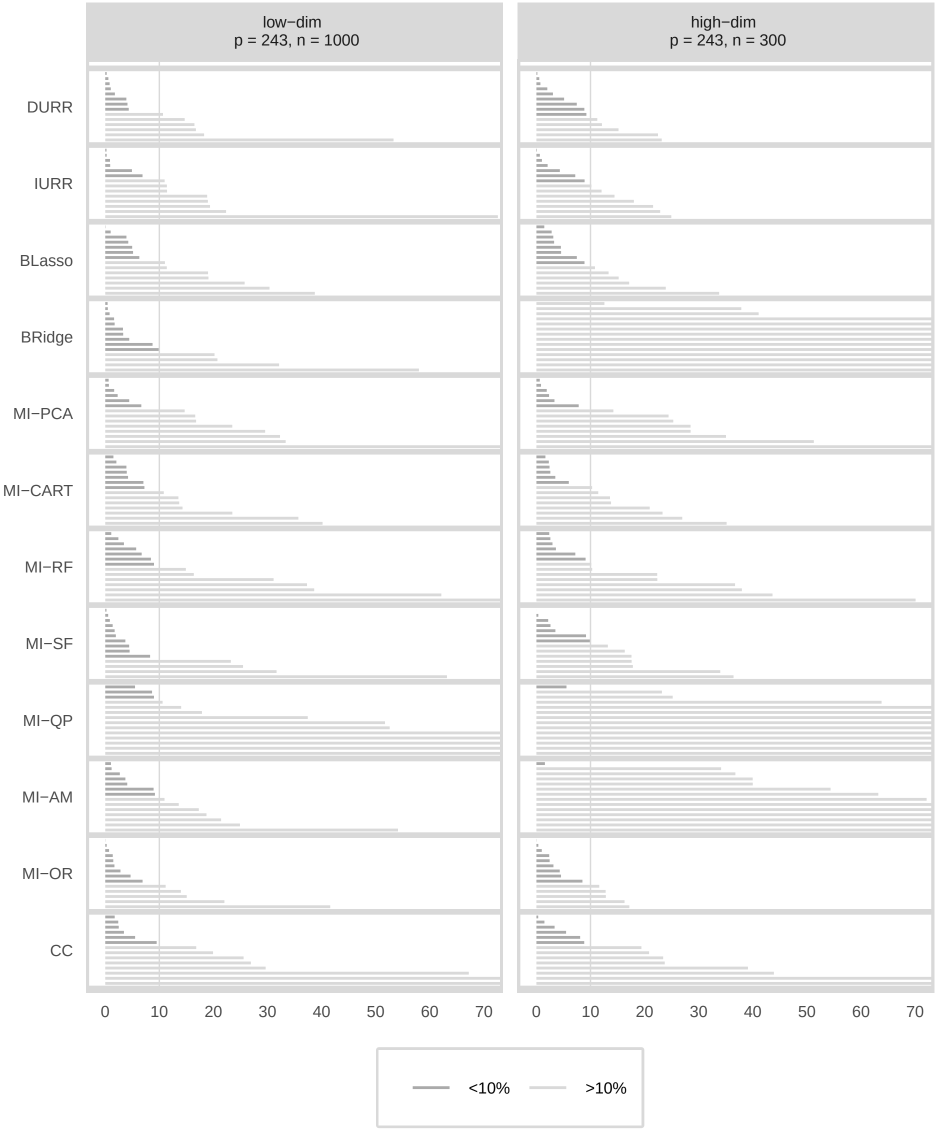


Figure 2.5: PRB for all the model parameters in model 2. For each method, the PRBs are ordered by increasing absolute value. The methods reported on the Y-axis are: indirect use of regularized regression (DURR), indirect use of regularized regression (IURR), MICE with Bayesian lasso (BLasso), MICE with Bayesian Ridge (BRidge), MICE with principal component analysis (MI-PCA), MICE with CART (MI-CART), MICE with random forests (MI-RF), MICE with step-forward selection (MI-SF), MICE with quickpred (MI-QP), MICE with analysis model (MI-AM), oracle low-dimensional MICE (MI-OR), and complete-case analysis (CC).

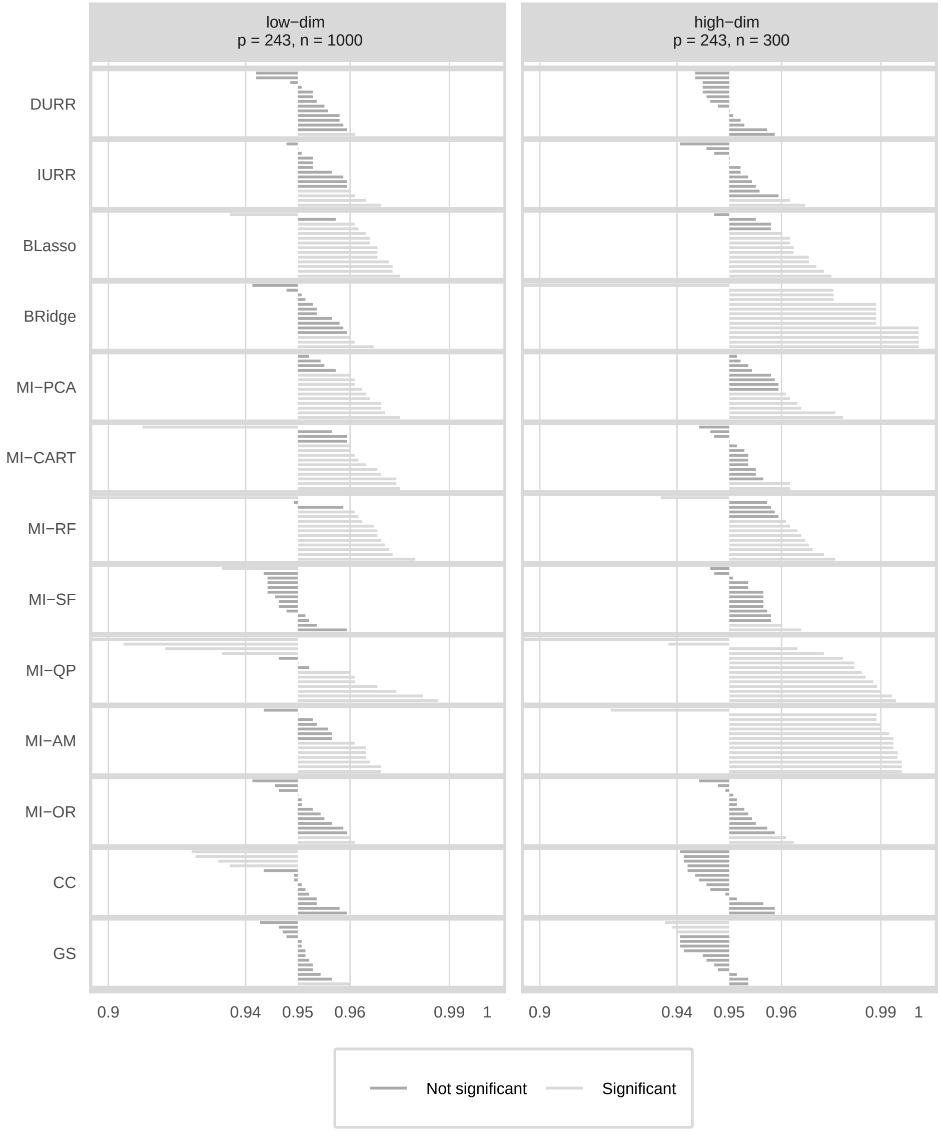


Figure 2.6: CIC for all model parameters in model 2. For each method, the CICs are ordered by increasing value. The methods reported on the Y-axis are: indirect use of regularized regression (DURR), indirect use of regularized regression (IURR), MICE with Bayesian lasso (BLasso), MICE with Bayesian Ridge (BRidge), MICE with principal component analysis (MI-PCA), MICE with CART (MI-CART), MICE with random forests (MI-RF), MICE with step-forward selection (MI-SF), MICE with quickpred (MI-QP), MICE with analysis model (MI-AM), oracle low-dimensional MICE (MI-OR), complete-case analysis (CC), and gold standard analysis (GS).

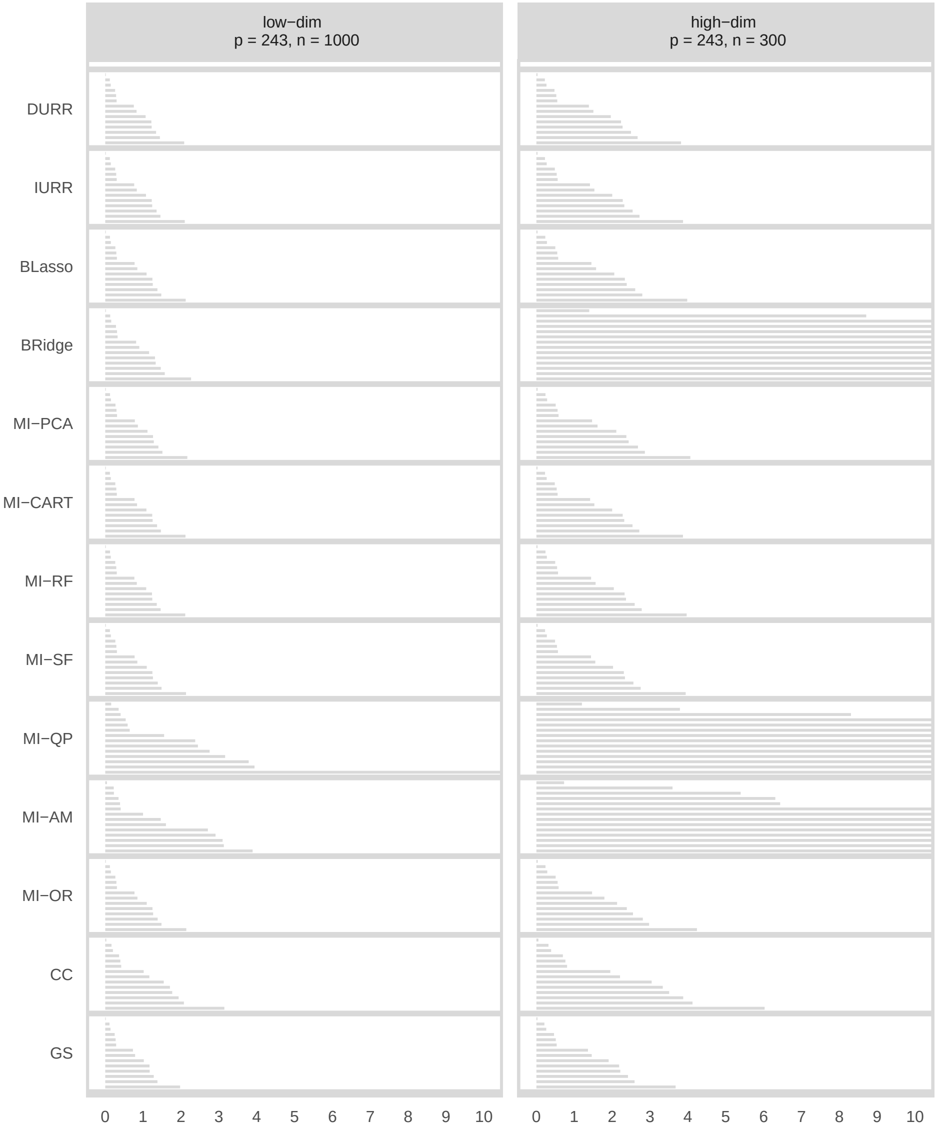


Figure 2.7: Average CIW for all model parameters in model 2. For each method, the CICs are ordered by increasing value. The methods reported on the Y-axis are: indirect use of regularized regression (DURR), Indirect Use of Regularized Regression (IURR), MICE with Bayesian lasso (BLasso), MICE with Bayesian Ridge (BRidge), MICE with Principal Component Analysis (MI-PCA), MICE with CART (MI-CART), MICE with random forests (MI-RF), MICE with step-forward selection (MI-SF), MICE with quickpred (MI-QP), MICE with analysis model (MI-AM), oracle low-dimensional MICE (MI-OR), complete-case analysis (CC), and gold standard analysis (GS).

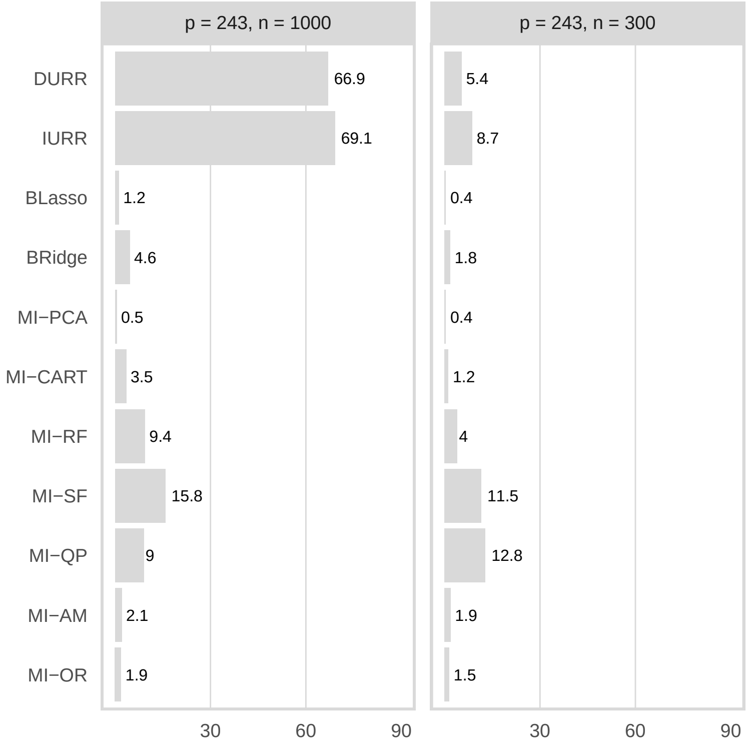


Figure 2.8: Average imputation time in minutes for the different MI methods when applied to the two different resampling study conditions.

estimation bias introduced by these two methods in the high-dimensional condition only slightly exceeded the PRB $= 10$ threshold, while most of the other MI methods resulted in covariance PRBs larger than 20 (with MI-PCA being the most salient exception). IURR and MI-SF produced good coverages in the low-dimensional condition but tended to under-cover in the high-dimensional condition, especially for variances and covariances. In the resampling study, IURR and MI-SF were also among the strongest performers. Although they did not demonstrate the best performance, there were no conditions in which IURR or MI-SF produced unacceptable results.

The confidence interval widths of IURR and MI-SF were in line with that of the other methods. In the simulation study, the confidence intervals produced by these methods were not influenced by the dimensionality of the data. In the resampling study, their confidence intervals were wider in the high-dimensional condition than in the low-dimensional one. However, this was the same pattern that affected most methods and it was caused by the smaller sample size we used to achieve the $p > n$

scenario in a data set with a fixed number of predictors. Overall, the confidence interval width pattern followed by IURR and MI-SF suggests that their imputation precision is not affected by a larger number of possible predictors.

From the end-user's perspective, IURR is an appealing method. IURR does not require the imputer to make choices regarding which variables are relevant for the imputation procedure. The only additional decision required of the imputer is selecting the number of folds to use when cross-validating the penalty parameter. As a result, an IURR imputation run is easy to specify, which makes IURR an appealing method for the imputation of large social scientific data sets. However, IURR is relatively computationally intensive. If the number of variables with missing values is large, IURR might result in prohibitive imputation time.

Similarly, an MI-SF run is easy to specify and only requires the user to choose the minimum sufficient increase in R^2 to use in the step-froward algorithm. However, the lack of clear guidelines on how to tune this parameter introduces more researcher's degrees of freedom than other methods. Finally, the imputation time of MI-SF was among the longest of the methods we considered.

In the simulation study, MI-PCA showed small bias and good coverage for both item means and covariances. Although it exhibited a large bias of the item variances, the—arguably more interesting—covariance relations between variables with missing values were always correctly estimated. Notably, MI-PCA was the only method resulting in small bias and close-to-nominal CIC for the covariances, even in the high-dimensional condition. When the CICs obtained with MI-PCA deviated significantly from nominal rates, they over-covered. In most situations, over-coverage is less worrisome than under-coverage as it leads to conservative, rather than liberal, inferential conclusions. In terms of confidence interval width, MI-PCA demonstrated the same pattern as IURR. In the resampling study, MI-PCA demonstrated middle-of-the-pack performance: somewhat worse than IURR, but still within acceptable levels.

In the additional simulation study evaluating the effects of collinearity, MI-PCA resulted in the same bias and confidence interval coverage as MI-AM when the potential auxiliary variables were highly correlated. This trend was caused by a subtle interaction between the data-generating model and the rule used to select the number of PCs. In every condition, there were only four true MAR predictors out of the pool of either 44 or 494 potential auxiliary variables. Consequently, the manner in which these four MAR predictors were represented in the component scores played a crucial role in the performance of MI-PCA. When ρ_{pav} was relatively small (i.e., the potential auxiliary variables were not strongly correlated), retaining enough components to explain 50% of the variance tended to select approximately 20 PCs.

Furthermore, the first of these components was predominately defined by the four MAR predictors, since these four variables comprised the entire subset of predictor data with non-trivial correlations. For high values of ρ_{pav}, however, the behavior of the MI-PCA algorithm shifted in two important ways. First, due to the increased homogeneity of the data, the first PC explained a much larger proportion of the total variance, so the 50% rule selected only one PC. Second, the first PC was predominately defined by the noise variables, since their high associations represented the majority of the reliable variance in the data. As a result, for large values of ρ_{pav}, the imputation models used by MI-PCA differed from the MI-AM imputation models only by adding a principal component that primarily summarized the noise variables as another useless predictor. A detailed explanation of this phenomenon is presented in module 3 of the interactive dashboard (Costantini, 2024c).

Importantly, this finding does not suggest that MI-PCA cannot treat highly collinear data. Rather, the poor performance seen here suggests that heuristic decision rules—such as keeping the first PC or enough components to explain 50% of the total variance—should not be mindlessly applied when running MI-PCA. Using a different non-graphical decision rule (e.g., the Kaiser criterion, Guttman, 1954; Kaiser, 1960) should preclude the problem described above and allow MI-PCA to compete with other automatic model-building strategies.

On balance, we believe the strong performance demonstrated by MI-PCA in the simulation study outweighs the mediocre performance shown in the resampling study. Furthermore, as noted above, the poor performance of MI-PCA in the high-collinearity study merely represents a weakness of our current implementation, not a general flaw in the underlying method. Consequently, we view MI-PCA as a promising approach for data analysts interested in testing theories on large social scientific data sets with missing values.

2.5.2 Methods with mixed results

In both the simulation study and the resampling study, BRidge manifested the same mixed performance. This method worked well when the imputation task was low-dimensional but led to extreme bias and unacceptable CI coverage in nearly all the high-dimensional conditions. Furthermore, the high-dimensionality of the data led to much wider confidence intervals compared to the ones obtained by other methods. Our results suggest that BRidge is effective only for low-dimensional imputation problems or in the presence of highly collinear data. The poor performance of BRidge compared to the other shrinkage methods might be explained by the fact that BRidge used a fixed ridge penalty across all iterations, while DURR, IURR, and BLasso allowed the penalty parameter to adapt to the improved imputations.

As implemented here, MI-QP was only effective in low-dimensional settings. The instability of MI-QP in high-dimensional scenarios was apparent not only because of its larger bias but also its very wide confidence intervals. The much wider 95% confidence intervals obtained by MI-QP in the high-dimensional scenario resulted in a 100% coverage, despite the large bias, revealing grossly imprecise imputations. MI-QP is also unable to address collinearity, as it selects predictors based on their bivariate relations with the variable under imputation and its missing data indicator without considering associations between the selected predictors. Hence, when faced with many highly correlated predictors, MI-QP can also be extremely computationally intensive due to the need to invert near-singular matrices.

DURR performed very well in the resampling study and quite poorly in the simulation study. In the resampling study, DURR was probably the best overall method in terms of bias and coverage, but it performed very badly in the high-dimensional condition of the simulation study. In the simulation study's low-dimensional condition, DURR produced small bias, good CI coverage, and similar CIW to IURR for item means and variances. However, compared to IURR, it suffered from greater deterioration in performance when applied to high-dimensional data, especially in terms of coverage. Our results suggest that DURR may have some unique benefits when treating the types of more discrete data seen in the resampling study. On balance, though, DURR probably should not be preferred to IURR.

There was little difference in performance between the use of CART and random forests as elementary imputation methods within the MICE algorithm. In line with what Doove et al. (2014) found, when a difference was noticeable, the simpler CART generally outperformed the more complex random forests. Both MI-CART and MI-RF produced large covariance bias in the simulation study. Although the bias for means and variances was acceptable, it was usually larger than that obtained by other MI methods. Furthermore, in terms of CI coverage, both methods showed a large under-coverage of the true values in the high-dimensional condition. In the resampling study, MI-CART and MI-RF both showed somewhat better performance than in the simulation study but not enough better to outweigh the mediocre simulation study performance. Although the nonparametric nature of these approaches elegantly avoids over-parameterization of imputation models, these methods were still outperformed by IURR and MI-PCA.

In the simulation study, BLasso resulted in small biases for item means and variances, even in the high-dimensional conditions, but it produced unacceptably biased covariance in both the low- and high-dimensional conditions. On the other hand, BLasso seemed to recover the relationships between variables in the resampling study well, where the overall bias levels for the regression coefficients were similar to those of MI-OR. However, in terms of CI coverage, BLasso showed poor performance in both studies resulting in either under-coverage or over-coverage for most parameters in the high-dimensional conditions.

The mixed performance of BLasso is also accompanied by a few obstacles to its application for social scientific research. Using Hans (2010b)'s Bayesian Lasso requires the specification of six hyper-parameters, which introduces more researcher degrees of freedom and demands a strong grasp of Bayesian statistics. Furthermore, the method has not currently been developed for multi-categorical data imputation, a common task in the social sciences. As a result, we do not recommend BLasso for the imputation of large social science data sets.

Finally, we do not recommend using MI-AM to impute large social science data sets. MI-AM bypasses the need to select which of the many potential auxiliary variables should be included in the imputation models by using only the analysis model variables as predictors. Therefore, MI-AM can be effective if the MAR predictors are part of the analysis model, but, as shown in the simulation study, it can lead to biased parameter estimates if they are not. In our simulation study, smaller biases and better coverages could always be achieved by using at least one of the alternative methods we evaluated.

2.6 Limitations and future directions

The present study aimed to compare current implementations of existing imputation methods. As a result, the scope of the simulation and resampling studies was limited by the current development state of the different methods. For example, DURR, IURR, and MI-PCA allow imputation of any type of data: DURR and IURR have been developed for categorical data imputation (Deng et al., 2016), and MI-PCA can be performed with any standard imputation model for categorical data. However, BLasso has not been formally developed for imputing multi-categorical variables yet. This limitation of BLasso forced us to work with missing values on variables that are either continuous or usually considered as such in practice (e.g., Likert-type scales). To maintain a fair comparison with BLasso, all methods were implemented with the assumption that the imputed variables are continuous and normally distributed. However, IURR, DURR, and MI-PCA could have performed differently in the resampling study if we had used their ordinal data implementations.

More generally, the results reported in this article only apply to the specific implementations of the algorithms we used. Many of the methods discussed could have been implemented differently. Zhao and Long (2016) proposed versions of IURR and DURR using the elastic net penalty (Zou & Hastie, 2005) and the adaptive lasso (Zou, 2006) instead of the lasso penalty. Although no substantial performance differences between penalty specifications emerged from the work of Zhao and Long (2016) or Deng et al. (2016), we must acknowledge that we did not investigate the impact of different types of regularization in the present study. Similarly, we have not investigated the sensitivity of BLasso to different hyper-parameters choices. Furthermore, the use of random forests within the MICE algorithm followed Doove et al. (2014), the version supported in the popular R package *mice*. However, Shah et al. (2014) independently developed another implementation of random forests within the MICE algorithm, which was available in the now archived R package *CALIBERrfimpute* (Shah, 2018). We are not aware of any evidence or theoretical reason to expect differences between the two implementations, but we did not verify this empirically. Finally, there are many alternatives to ordinary least square (OLS) estimation that we did not consider. Dempster et al. (1977) compared the properties of 57 such OLS alternatives, including different variants of ridge regression, subset regression (e.g., forward and backward model selection), and principal component regression, when applied to fully observed data. Any of these variants could be used as the elementary imputation model in a MICE implementation. In the present study, however, our inclusion criteria for imputation methods precluded consideration of these alternatives. We considered only those high-dimensional prediction methods that have already been recommended in the literature specifically for MI. This is the same reason we did not consider many state-of-the-art prediction methods like (deep) neural networks or support vector machines/regressions, even though those methods currently dominate all others in terms of raw prediction and classification performance.

Our implementation of MI-PCA was limited in several ways. First, MI-PCA requires choosing the number of components to extract from the auxiliary variables. In this study, we decided to retain the first components that explained 50% of the total variance in the auxiliary variables. However, this decision was arbitrary, and the results of collinearity-focused simulation study clearly demonstrate some of the possible deleterious consequences of this approach. Additionally, the good performance of MI-PCA may have been partially driven by the fact that, while imputing the jth variable, all other variables under imputation were used directly as predictors. If the other variables under imputation had been included in the imputation models through the PCs extraction step, and not used as separate, individual predictors, the performance of MI-PCA might have been less favorable. In Costantini et al. (2024) we assess the effects of these two factors on the MI-PCA method. The unsupervised nature of the classical PCA through which MI-PCA constructs imputation model predictors may also be a limiting feature. While classical PCA should optimally distill the variance of the potential auxiliary variables into a succinct set of component scores, these component scores may not be useful predictors in the imputation model (e.g., if most of the potential auxiliary variables were not good predictors to begin with). Supervised versions of PCA (e.g., supervised PCA, Bair et al., 2006, principal covariates regression, de Jong and Kiers, 1992) could overcome this limitation. In Costantini et al. (2024) we evaluate the performance of MI-PCA when the component scores are extracted via several different supervised versions of PCA.

2.7 Conclusions

Our objective in this project was to find a good data-driven way to select the predictors that go into an imputation model. A wide range of methods have been proposed to address this issue, but little research has been done to compare their performance. With this article, we start to fill this gap and provide initial insights into applying such methods in social science research. IURR, MI-SF, and MI-PCA showed promising performance when compared to other high-dimensional imputation approaches. While all of these methods represent good options for automatically defining the imputation models of an MI procedure, MI-PCA is the more practically appealing option due to its much greater speed. However, the current implementation of MI-PCA is limited, and making the most of this method will require further research and optimization, especially regarding methods for the number of components. Finally, Bayesian ridge regression is a good alternative when the imputer wants to have an automatic way of defining the imputation models in a low-dimensional setting ($n \gg p$).

Chapter 3

Solving the many-variables problem in MICE with principal component regression

Abstract Multiple Imputation (MI) is one of the most popular approaches to addressing missing values in questionnaires and surveys. MI with multivariate imputation by chained equations (MICE) allows flexible imputation of many types of data. In MICE, for each variable under imputation, the imputer needs to specify which variables should act as predictors in the imputation model. The selection of these predictors is a difficult, but fundamental, step in the MI procedure, especially when there are many variables in a data set. In this project, we explore the use of principal component regression (PCR) as a univariate imputation method in the MICE algorithm to automatically address the many-variables problem that arises when imputing large social science data. We compare different implementations of PCR-based MICE with a correlation-thresholding strategy through two Monte Carlo simulation studies and a case study. We find the use of PCR on a variable-by-variable basis to perform best and that it can perform closely to expertly designed imputation procedures.

Results dashboard To run the results dashboard accompanying this chapter install the Shiny app as an R package from the Zenodo permanent repository:

```r
# Install shiny app
devtools::install_url(
        "https://zenodo.org/records/10759108/files/EdoardoCostantini/plotmipca-v2.2.zip"
)
```

Then, you can start the app by running this command:

```r
# Start the app
plotmipca::start_app()
```

Reproducibility statement The R code used to produce the results reported in this chapter can be found in the permanent repositories stored on Zenodo (for the simulation study see Costantini 2023d, for the case study see Costantini 2023a). The README.md file contains instructions on how to reproduce the results.

Costantini, E., Lang, K. M., Sijtsma, K., & Reeskens, T. (2024). Solving the many-variables problem in MICE with principal component regression. *Behavior Research Methods*, *56*(3), 1715–1737. https://doi.org/10.3758/s13428-023-02117-1

3.1 Introduction

Missing values are a problem afflicting virtually all data sets in the social and behavioral sciences. Multiple Imputation (MI) is one of the most popular approaches to address the issue of non-response. Although MI can treat essentially any missing data problem, it was originally designed to impute large surveys, especially when those surveys are used to create publicly released data that many researchers will analyze independently (Rubin, 1996). In this context, MI was envisioned as being especially useful when the data collector (and imputer) is distinguished from the ultimate user (or analyst).

The imputer's main task is to define an imputation model that supports analyses from many users. A well-designed imputation model should include all the predictors of missingness present in the data, and it should incorporate all the features of the substantive analysis models that will be used by the data analysts. If important predictors of missingness are left out of the imputation model, the missing at random (MAR) assumption is violated (Collins et al., 2001, p. 339). If some features of the substantive analysis model of interest do not appear in the imputation model, the two models are said to be uncongenial, a situation that can invalidate the inferential conclusions obtained after imputation (Meng, 1994; R. J. A. Little and Rubin, 2002, p. 218).

To decide which predictors to include in the imputation model, a commonly recommended strategy is to include as many predictors as possible (i.e., the inclusive strategy, Collins et al., 2001). However, the scale of modern social surveys and data collection endeavors complicates the task of selecting these predictors. Cross-sectional social surveys (e.g., World values survey, Haerpfer et al., 2020; European values study, EVS, 2020a) commonly measure hundreds of variables, and including all of these variables in the imputation model can lead to prohibitively long imputation times and convergence failures (van Buuren, 2018, p. 259). Social and behavioral scientists also frequently work with longitudinal surveys and panel studies (e.g., Panel Study of Income Dynamics, McGonagle et al., 2012; LISS Panel, Scherpenzeel and Das, 2018), which can lead to data sets with many more columns than rows. This high-dimensionality can result in singularity issues (Hastie et al., 2009, p. 46) when estimating the imputation models. The imputer needs to address this many-variables problem by thoroughly scanning all the available variables to decide which of them should be used in the imputation models. In this article, we explore the use of principal component regression to automate the definition of the imputation model by replacing a large number of possible predictors with a small subset of principal components (PCs).

3.1.1 MICE and the many-variables problem

In social science research, multivariate imputation by chained equation (MICE, van Buuren & Groothuis-Oudshoorn, 2011) has been implemented in all major statistical software (e.g., Stata, StataCorp, 2013; SPSS, IBM Corp., 2020; R, van Buuren and Groothuis-Oudshoorn, 2011) and is arguably the most popular way to implement MI. MICE, also known as fully conditional specification and sequential regression imputation (Raghunathan et al., 2001), is an iterative algorithm that obtains imputations from the implied multivariate distribution of the missing data by sampling from a set of univariate conditional densities. This algorithm requires the definition of a conditionally specified univariate imputation model for each variable under imputation. At every iteration, each univariate imputation model is used to obtain replacement values for the missing data points. When convergence is reached, any sample from the univariate imputation models' predictions represents a sample from the target multivariate data distribution. These samples are used to define multiple versions of the original data, with different plausible values used to replace the original missing data. Any analysis model of scientific interest can then be estimated on each of the multiply imputed data sets. The estimates of the parameters of interest in the analysis model are then pooled following Rubin's rules (Rubin, 1987, p. 76).

The definition of the univariate imputation models is a fundamental step for the good performance of the MICE procedure. For each variable under imputation, the imputer needs to define a univariate imputation model. This task involves two decisions:

1. Selecting the model form;
2. Selecting the predictors.

The first decision is usually guided simply by the measurement level of the variables under imputation. For example, continuous variables can be imputed using a linear regression model, while binary variables can be imputed using logistic regression. The second decision requires choosing the variables to be included as predictors in the imputation model. In general, it is advisable to adopt an inclusive strategy (Collins et al., 2001), meaning including as many predictors as possible in the univariate imputation models. Using as much information as possible from the data leads to multiple imputations that have minimal bias and maximal efficiency (Collins et al., 2001; Meng, 1994). Furthermore, including more predictors in the univariate imputation models makes the MAR assumption more plausible (Collins et al., 2001, p. 339). Finally, if the imputation model omits variables that are part of the analysis model that will be estimated on the imputed data, the analysis model's parameter estimates might be biased (Enders, 2010, p. 229), and the attendant confidence intervals might be too wide (R. J. A. Little & Rubin, 2002, p. 218). As a result, including more predictors in the imputation models increases the range of analysis models that can be estimated with a given set of imputations (Meng, 1994).

When a data set consists of only a few variables (i.e., tens of variables), it may be feasible to include all of these variables in all the univariate imputation models. However, standard imputation methods face computational limitations in the presence of a large number of predictors (i.e., hundreds). For example, MICE using Bayesian imputation under the normal linear model (van Buuren, 2018, p. 68) requires the number of predictors (p) in the univariate imputation model to be smaller than the number of observed cases (n) to avoid computational problems with the system of equations (James et al., 2013, p. 203). Even when the number of predictors is smaller than the number of observations, including many predictors in the imputation models can increase the chances of collinearity issues (van Buuren, 2018, pp. 167–170) and can bias the analysis model parameter estimates (Hardt et al., 2012).

The type of data social and behavioral scientists work with today often contains many variables. For example, a single wave of the World Values Survey (Haerpfer et al., 2020) contains more than 300 variables, and a single wave of the European Values Study (EVS, 2020a) contains around 250 variables. Running a MICE algorithm on this type of data, without selecting a subset of variables to use as predictors in the univariate imputation models, requires the algorithm to estimate regression models with hundreds of predictors for each variable under imputation. With such a specification, the algorithm will be extremely slow, and the imputations will usually be poor. However, selecting a smaller subset of predictors for each univariate imputation model can be a daunting task. Choosing which predictors should be included in the univariate imputation models that constitute a run of the MICE algorithm entails a considerable degree of subjective judgment and requires both statistical and substantive expertise to achieve satisfactory results.

Van Buuren (2018, pp. 270–271) provides a summary of different strategies an expert imputer can employ when designing imputation models for social science data sets with many variables. The imputer can:

1. *Remove constants and collinear variables.* Collinear predictors will lead to unstable imputation model parameter estimates. Using one of a set of collinear

predictors reduces the size of the predictor space for imputation models without losing any important information.

2. *Evaluate statistics describing the connection between variables in the data.* For example, one can compute the proportion of usable cases for imputing a variable based on another (i.e., inbound statistic van Buuren, 2018, p. 108). The more cases that are usable, the more *connected* the two variables are. The influx-outflux coefficients (van Buuren, 2018, pp. 109–111) provide overall measures of how each variable *connects* to the rest of the data. In general, variables with high influx and outflux are preferred as predictors in an imputation model.

3. *Apply a correlation-thresholding strategy.* Only variables that are associated with the variables under imputation can be effective predictors in the imputation models. As a result, an intuitive strategy to select a small number of important predictors is to include only variables that correlate with the ones under imputation more strongly than a chosen threshold. However, the optimal threshold is not obvious. While choosing a low threshold might lead to selecting too many variables, choosing a high threshold might lead to excluding important predictors.

These strategies are not guaranteed to avoid over-parameterization of the univariate imputation models, and more complex (combinations of) strategies are often needed. The nature of some social science data sets offers a few other opportunities to reduce the dimensionality of the imputation models. For example, with longitudinal data sets, the imputer might decide to use only the first measurement of the same construct when imputing other variables, or she may use the total score in place of the many items constituting a scale. Additionally, an imputer can use high-dimensional prediction methods as univariate imputation models. Shrinkage methods, non-parametric prediction algorithms, and dimension reduction techniques can all be incorporated into MICE to reduce the complexity of the predictor selection step.

Zhao and Long (2016) and Deng et al. (2016) proposed the use of lasso regression (Tibshirani, 1996) within MICE. Lasso is a shrinkage technique that can provide a data-driven selection of important predictors for each imputation model. Decision trees are a popular class of semi-parametric prediction algorithms that can accommodate many predictor variables and represent complex, nonlinear relations among the variables (Burgette & Reiter, 2010; Doove et al., 2014; Shah et al., 2014). Decision trees have already been integrated into popular imputation software (e.g., the R package *mice*, van Buuren & Groothuis-Oudshoorn, 2011). Howard et al. (2015) proposed using principal component analysis (PCA; Jolliffe, 2002, pp. 1–6) to reduce a set of auxiliary variables into a small set of principal components (PCs). By extracting PCs from the (potentially numerous) auxiliary variables, this method can summarize the information contained in the auxiliary variables with just a few component scores. These PCs can then be used as predictors in a standard, low-dimensional application of MICE.

The approaches described above have the potential to automatically address many of the issues caused by having too many variables available as potential imputation model predictors. By default, high-dimensional prediction methods avoid collinearity issues and hence stabilize imputation model estimation. Furthermore, many high-dimensional methods offer some form of variable selection or dimensionality reduction that can reduce the burden of making predictor choices. Finally, algorithms that incorporate some form of regularization or dimension reduction allow the imputer to include more predictors in their imputation model, thereby increasing the chances of satisfying the MAR assumption.

A bespoke application of the MICE algorithm driven by subject-matter expertise may lead to better imputations than using automatic, data-driven approaches (though, as we illustrate in Section 3.5, this need not always be the case). However, expert knowledge is not always available for every project that could benefit from imputation. Furthermore, high-dimensional prediction models do not need to be the sole solution to the many-variables imputation problem. These methods can always be combined with expert knowledge to further improve the quality of imputation.

Costantini, Lang, Reeskens, and Sijtsma (2023) compared a wide range of high-dimensional MI approaches, including the use of (Bayesian) lasso, ridge regression, random forests, correlation-thresholding, and PCA within the mice algorithm. They found that using frequentist lasso to select the imputation model predictors and using PCA to reduce the dimensionality of the imputation models produced the best results. The PCA-based approach was the strongest overall performer, though. Incorporating PCA into the MICE algorithm consistently led to small estimation bias and close-to-nominal confidence interval coverage for the analysis model parameters. However, that study considered only a single implementation of PCA that was applied in a limited set of conditions. In this paper, we extend the findings of Costantini, Lang, Reeskens, and Sijtsma (2023) by further investigating the use of PCA within the MICE algorithm. We use Monte Carlo simulation studies and a real-data case study to compare the performance of three alternative PCA-based MI approaches and evaluate how certain data characteristics impact that performance.

Two of these PCA-based MI approaches have been previously described in the literature. We propose a novel third method here.

3.1.2 Principal component analysis for MICE

PCA is a dimensionality reduction technique by which a set of variables is summarized with a smaller number of PCs. These PCs are defined such that they explain the largest possible proportion of the original data's variance, given the number of PCs. PCA can be used in conjunction with many statistical techniques, and its use in regression analysis has been extensive (e.g., de Jong & Kiers, 1992; S. Park et al., 2021; Reiss & Ogden, 2007; Rosipal et al., 2001). In particular, one of the best-known uses of PCA in multiple regression is principal component regression (PCR; Jolliffe, 2002, pp. 168–173), where PCs act as predictors in a multiple regression model.

Standard implementations of the MICE algorithm cycle through a sequence of univariate imputation models (i.e., one model for each incomplete variable). Any, or all, of these univariate imputation models can be replaced by PCR. We refer to this use of PCR in conjunction with MICE as MI-PCR. The MI-PCR method is a broad approach that can be implemented in many different ways. For example, a single set of PCs could be estimated before running MICE. These PCs could then be used in a subsequent run of MICE as imputation model predictors. Alternatively, a new PCA could be run within every iteration of MICE to produce updated PCs that incorporate the information from the most recent imputations. These updated PCs could then be used as predictors to generate the imputations for that single iteration. In this report, we primarily wish to investigate how different implementations of MI-PCR impact imputation quality.

We also explore how the number of components used as predictors in MI-PCR influences imputation quality. Based on their analysis of data with a simple unidimensional latent structure, Howard et al. (2015) proposed retaining only the first PC[12]. However, medical and social science data are frequently characterized by complex latent structures that are unlikely to be well-summarized by a single component. Therefore, evaluating the impact of the number of PCs is a secondary purpose of the present study. Finally, as a tertiary focus, we also explore how key characteristics of the data affect MI-PCR. In particular, we evaluate how the measurement level

---

[12] Since the first component explained 40% of the variance in their simulations, Howard et al. (2015) alternatively recommended using the minimum number of components necessary to explain 40% of the variance in the auxiliary variables.

of the potential predictors and their strengths of association impact the performance of MI-PCR.

In this manuscript, we present the results of two Monte Carlo simulation studies through which we explore the performance of MI-PCR. We assess this performance based on the estimation bias, confidence interval width, and confidence interval coverage (see Section 3.3.1.4 for details). We also apply the different implementations of MI-PCR to the Fireworks Disaster data (i.e., a real clinical psychology data set that has previously been used to demonstrate high-dimensional imputation problems, van Buuren, 2018, p. 313). In what follows, we describe how PCA can be used within the MICE algorithm (Section 3.2). We then discuss the simulation studies and the case study in Sections 3.3, 3.4, and 3.5, respectively. We provide a general discussion in Section 3.6, and share some final remarks on the selection of the number of components in Section 3.7. We discuss limitations and future directions in Section 3.8. Finally, we state our conclusions in Section 3.9.

3.2 MI-PCR: MICE using PCR

Here we briefly describe the MICE algorithm, PCA, and how PCA can be used in conjunction with MICE. We use the following notation. Scalars are denoted by lowercase letters in light typeface (non-bold). Vectors and matrices are denoted by bold lowercase and bold uppercase letters, respectively. We use the subscripts ‘obs’ and ‘mis’ to refer to the observed and missing elements in a vector. For a given data set, we refer to the variables that are part of the researcher’s model of scientific interest (e.g., the linear regression used to answer a research question) as the analysis model variables. We refer to all other variables as potential auxiliary variables for the imputation models. We use the subscripts ‘am’ and ‘av’ to refer to variables that are part of either the analysis model or the set of potential auxiliary variables, respectively.

3.2.1 Multivariate imputation by chained equations

Consider an $n \times p$ data set $\mathbf{X}$. Its columns, $\mathbf{x}_1, \dots, \mathbf{x}_p$, represent variables, and the rows represent observational units (e.g., people participating in a social survey). Assume the first t columns of $\mathbf{X}$ have missing values. For each partially observed $\mathbf{x}_j$, with $j = 1, \dots, t$, the imputer defines a univariate imputation model:

$$f(\mathbf{x}_j|\mathbf{X}_{-j}, \theta_j), \tag{3.1}$$

where $\mathbf{X}_{-j}$ is the collection of variables in $\mathbf{X}$ excluding $\mathbf{x}_j$, and θ_j is a vector of imputation model parameters. Model (3.1) is usually a generalized linear model chosen according to the measurement level of $\mathbf{x}_j$. The MICE algorithm starts with replacing the missing values in each $\mathbf{x}_j$ with initial guesses. Then, at every iteration, each variable is imputed by its univariate imputation model. First, the imputation model parameters are drawn from their fully conditional posterior distributions, and then imputations are drawn from the posterior predictive distribution of $\mathbf{x}_j$.

For the jth variable under imputation at iteration k, the algorithm draws from the following distributions:

$$\theta_j^{(k)} \sim f(\theta_j) f(\mathbf{x}_{j,\text{obs}} | \mathbf{X}_{-j}^{(k)}, \theta_j), \tag{3.2}$$

$$\mathbf{x}_{j,\text{mis}}^{(k)} \sim f(\mathbf{x}_{j,\text{mis}} | \mathbf{X}_{-j}^{(k)}, \theta_j^{(k)}) \tag{3.3}$$

Equation (3.2) is the fully conditional posterior distribution defined as the product of $f(\theta_j)$, a prior distribution for θ_j, and $f(\mathbf{x}_{j,\text{obs}} | \mathbf{X}_{-j}^{(k)}, \theta_j)$, the likelihood of observing $\mathbf{x}_{j,\text{obs}}$ under the imputation model for $\mathbf{x}_j$. Equation (3.3) is the posterior predictive distribution from which updates of the imputations are drawn. In both equations, $\mathbf{X}_{-j}^{(k)}$ is $(\mathbf{x}_1^{(k)}, \dots, \mathbf{x}_{j-1}^{(k)}, \mathbf{x}_{j+1}^{(k-1)}, \dots, \mathbf{x}_p^{(k-1)})$, meaning that at all times the most recently imputed values of all variables are used to impute other variables.

Each iteration comprises one complete cycle through all t variables under imputation. After a sufficient number of iterations, the algorithm converges to a stable equilibrium, and the imputations represent samples from the target multivariate distribution. With this process, one can generate as many imputed data sets as desired. Finally, the analysis model is estimated on each imputed data set, and the parameter estimates are pooled using Rubin's rules (Rubin, 1987).

3.2.2 Principal component analysis

PCA finds a low(er)-dimensional representation of $\mathbf{X}$ with minimal loss of information. We refer to this low-dimensional representation as the $n \times q$ matrix $\mathbf{Z}$, where $q < p$. The columns of $\mathbf{Z}$ are called the principal components (PCs) of $\mathbf{X}$. The first PC of $\mathbf{X}$[13] is the linear combination of the columns of $\mathbf{X}$ with the largest variance:

$$\mathbf{z}_1 = \mathbf{x}_1 w_{11} + \mathbf{x}_2 w_{12} + \cdots + \mathbf{x}_p w_{1p} = \mathbf{X}\mathbf{w}_1, \tag{3.4}$$

with $\mathbf{w}_1$ being the $p \times 1$ vector of weights $w_{11}, \dots, w_{1p}$. The second principal component ($\mathbf{z}_2$) is defined by finding the vector of weights $\mathbf{w}_2$ giving the linear combination

---

[13] We follow the common practice of assuming that the columns of $\mathbf{X}$ are mean-centered and scaled to have a variance of 1.

of $\mathbf{x}_1, \dots, \mathbf{x}_p$ with maximal variance out of all the linear combinations that are uncorrelated with $\mathbf{z}_1$. Every subsequent column of $\mathbf{Z}$ can be understood in the same way: for example, $\mathbf{z}_3$ is the linear combination of $\mathbf{x}_1, \dots, \mathbf{x}_p$ that has maximal variance out of all the linear combinations that are uncorrelated with $\mathbf{z}_1$ and $\mathbf{z}_2$. As a result, all PCs are uncorrelated by definition and every subsequent PC has a lower variance than the preceding one. We can write the relationship between all the PCs and $\mathbf{X}$ in matrix notation:

$$\mathbf{Z} = \mathbf{X}\mathbf{W}, \tag{3.5}$$

where $\mathbf{W}$ is a $p \times q$ matrix of weights, with columns $\mathbf{w}_1, \dots, \mathbf{w}_q$. Equation (3.5) also allows us to understand PCA as the process of projecting the original data from a p-dimensional space to a q-dimensional space. The weight vectors $\mathbf{w}_1, \dots, \mathbf{w}_q$ define the directions in which the n observations of $\mathbf{x}_1, \dots, \mathbf{x}_p$ are projected. The projected values are the principal component scores $\mathbf{Z}$.

3.2.3 Principal component regression

PCR replaces the p predictors of a regression model with q PCs extracted from those predictors. Given the data $\mathbf{X}$, consider a standard regression model where the jth variable is regressed on the other columns in the data:

$$\mathbf{x}_j = \mathbf{X}_{-j}\beta + \epsilon, \tag{3.6}$$

where $\mathbf{x}_j$ is a $n \times 1$ vector of dependent variable scores, β is a $(p-1) \times 1$ vector of $p-1$ regression coefficients, and ϵ is a $n \times 1$ vector of independent normally distributed errors. With PCR we use the PCs of $\mathbf{X}_{-j}$ in place of $\mathbf{X}_{-j}$ in the regression model so that Equation (3.6) can be rewritten as:

$$\mathbf{x}_j = \mathbf{Z}\gamma + \epsilon, \tag{3.7}$$

where γ is a $q \times 1$ vector of regression coefficients. The lower dimensionality of $\mathbf{Z}$ compared to $\mathbf{X}_{-j}$, and the independence of its columns, allow Equation (3.7) to address the computational limitations of Equation (3.6) in the presence of many predictors.

3.2.4 MI-PCR

Standard MICE formulations are based on univariate imputation models that suffer from the same computational limitations that we discussed for multiple regression. In general, the idea of MI-PCR is to use Equation (3.7) as the univariate imputation model for every variable under imputation. By doing so, we aim to address the many-variables problem by summarising the imputation models' predictors with just a few PCs. However, this idea can be implemented in many ways. PCA can be used at different stages of the MICE algorithm, and different sets of variables can be summarized by the PCs. In the following sections, we describe the three implementations evaluated in this study.

3.2.4.1 MICE with PCA on auxiliary variables

The most straightforward way to use PCA within MICE is to compute a single set of PCs based only on the potential auxiliary variables. In general, potential auxiliary variables include predictors of missingness, variables related to the ones under imputation, and variables that are useless for imputation. To reduce the dimensionality of imputation models, an expert imputer would usually examine these variables to locate and exclude members of the last group from the imputation models. PCA can be used as an alternative, data-driven pre-processing step to project all the potential auxiliary variables onto a lower-dimensional space and bypass the need to select which variables to use as predictors in the imputation models. We refer to this approach as MICE with PCA on the auxiliary variables (MI-PCR-AUX).

In MI-PCR-AUX, the univariate imputation models use as predictors the raw version of any variable that is part of the analysis model, and the principal components summarizing the potential auxiliary variables. We can write the univariate imputation model as:

$$f(\mathbf{x}_j|\mathbf{X}_{\text{am},-j}, \mathbf{Z}_{\text{av}}, \theta_j), \tag{3.8}$$

where $\mathbf{X}_{\text{am},-j}$ is the set of analysis model variables except the one under imputation, and $\mathbf{Z}_{\text{av}}$ is the set of PCs estimated from the set of potential auxiliary variables $\mathbf{X}_{\text{av}}$. This use of PCA was proposed by Howard et al. (2015).

The strength of MI-PCR-AUX is using the raw analysis model variables (not filtered via the PCA) while including as much auxiliary information as possible (filtered via the PCA). However, MI-PCR-AUX requires knowledge of the analysis model before imputation, and the possible presence of missing values in the potential auxiliary variables needs to be addressed. Howard et al. (2015) suggested using single imputation to create a complete set of potential auxiliary variables, but implementing this idea is not necessarily straightforward. All the obstacles to defining the univariate imputation models previously discussed still arise during this single imputation procedure.

3.2.4.2 MICE with PCA on all variables

One way to relax the requirements of knowing the analysis model before running the imputation procedure is to extract PCs from all available variables—including the ones under imputation—and then use only these PCs as predictors in the imputation models. This approach (hereafter, MI-PCR-ALL) is implemented in the R package *PcAux* (Lang et al., 2018). The univariate imputation model for this approach can be written as:

$$f(\mathbf{x}_j|\mathbf{Z}, \theta_j), \tag{3.9}$$

where $\mathbf{Z}$ is the set of PCs estimated on $\mathbf{X}$. Ideally, MI-PCR-ALL supports a wide range of analysis models, as the information on every available variable is summarized by the PCA procedure and included in all the imputation models. In theory, the imputer could even augment $\mathbf{X}$ before extracting the PCs with every desired interaction and polynomial term that might be present in an analysis model.

As noted above, PCA cannot be performed in the presence of missing values. Yet, to implement MI-PCR-ALL, we must perform PCA on all the variables in $\mathbf{X}$—even the ones targeted by imputation. So, we must first (temporarily) treat the missing data to allow the PC extraction. Our implementation of MI-PCR-ALL begins by filling in the missing values with a single imputation and extracting PCs from this completed data set. It is important to note that these imputations will not be used for statistical inference. So, the attenuated standard errors known to result from single imputation are not a concern. As long as the imputations are well-constructed and consistent with the distribution of the original variables, inference based on data imputed with MI-PCR-ALL should not be negatively impacted. Nevertheless, the performance of MI-PCR-ALL is tied to the quality of this first single imputation.

In MI-PCR-ALL, the PCs are the only predictors used in the univariate imputation models and do not have missing values. Therefore, a single iteration of the MICE algorithm is sufficient. Computationally, this is an advantage as there is no need to perform any burn-in iterations.

3.2.4.3 MI with PCA on a variable-by-variable basis

The most flexible way to incorporate PCA into MICE is to extract PCs at every iteration. When imputing $\mathbf{x}_j$ at the kth iteration, PCs can be estimated from $\mathbf{X}_{-j}^{(k)}$ and used as predictors in the univariate imputation model. Each univariate imputation model can then be defined as:

$$f(\mathbf{x}_j|\mathbf{Z}_{-j}^{(k)}, \theta_j), \tag{3.10}$$

where $\mathbf{Z}_{-j}^{(k)}$ is the matrix storing the PC scores estimated on $\mathbf{X}_{-j}^{(k)}$. We refer to this approach as MI-PCR-VBV because of the variable-by-variable use of PCA.

As with MI-PCR-ALL, MI-PCR-VBV does not require knowledge of the analysis model prior to imputation and it can support a wide range of analysis models. Moreover, by extracting PCs at every iteration from variables with the most recently imputed values, MI-PCR-VBV addresses missing values in one step, without requiring a pre-processing single imputation procedure. The disadvantage of MI-PCR-VBV is in the higher computational intensity relative to both MI-PCR-AUX and MI-PCR-ALL. Performing PCA on large social surveys involves demanding matrix operations. MI-PCR-VBV requires repeating these intensive manipulations for every variable under imputation and for every iteration of the MICE algorithm.

3.3 Simulation study 1

We investigated the relative performance of the methods described above with a Monte Carlo simulation study. In particular, we were interested in assessing the estimation bias, confidence interval width, and confidence interval coverage of statistics estimated from the imputed data. The purpose of this study was to evaluate these statistical properties of MI-PCR in several settings that differed in the proportion of noise variables present in the data, the measurement level of the variables, and the number of PCs used in the imputation models.

When defining the univariate imputation model, an expert imputer would usually exclude all variables that are weakly associated with the variables under imputation. We refer to these weak predictors of the imputation targets as noise variables. When using MI-PCR, noise variables will contribute to the construction of the PCs as much as the important predictors of the variables under imputation. Consequently, PCs extracted from data that contain a large proportion of noise variables may be more weakly associated with the variables under imputation. We expect that the presence of a larger proportion of noise variables will negatively impact the performance of MI-PCR (i.e., larger bias, lower efficiency, larger deviation from nominal coverage). Additionally, in real survey applications, theoretical constructs of interest are often measured with discrete items such as Likert scales. The number of categories with which information is recorded in a variable can impact on how well $\mathbf{Z}$ represents $\mathbf{X}$. Finally, each of the implementations of MI-PCR described above can be used with different numbers of PCs. Howard et al. (2015) suggested that using the first PC may be sufficient. However, they used a set of strongly associated potential auxiliary variables measuring a single latent factor. When the underlying correlation structure is more complex (i.e., more than one latent factor, different correlation levels) using only the first PC is likely to result in a poor representation of the data and poor imputation performance. In what follows we outline the simulation study procedure, discuss the experimental factors in detail, and report the results.

3.3.1 Method

The simulation study procedure involved five steps:

1. Data generation: We simulated $S = 500$ data sets from a confirmatory factor analysis model, following the procedure described in Section 3.3.1.1.
2. Missing data imposition: We imposed missing values on four target items in each generated data set, following the procedure described in Section 3.3.1.2.
3. Imputation: For each incomplete data set, we applied each of the different imputation methods to generate d multiply imputed data sets, as described in Section 3.3.1.3.
4. Analysis: We used the d imputed data sets to estimate the means, variances, covariances, and correlations of the four items with missing values, and we pooled the estimates according to Rubin's rules (Rubin, 1986, p. 76).
5. Evaluation: We assessed the performance of each imputation method by computing the bias, confidence interval width, and confidence interval coverage of the above statistics, as described in Section 3.3.1.4.

3.3.1.1 Data generation

For each replication, we generated a 500×56 matrix of fully observed data $\mathbf{X}$. We fixed the sample size to 500 observations to generate data sets that would have statistical properties similar to large social science data sets without needlessly increasing the computational demands of the simulation study. Each data set was generated based on the following confirmatory factor analysis model:

$$\mathbf{X} = \mathbf{F}\mathbf{\Lambda}' + \mathbf{E}, \tag{3.11}$$

where $\mathbf{F}$ is a 500×7 matrix of latent variables scores, $\mathbf{\Lambda}$ is a 56×7 matrix of factor loadings, $\mathbf{\Lambda}'$ is its transpose, and $\mathbf{E}$ is a 500×56 matrix of measurement errors. The dimensionality of the data resembles that of short-scale questionnaires used in the social sciences. For example, consider the NEO Five-Factor Inventory (Costa

Jr & McCrae, 2008, NEO-FFI), which measures the Big Five personality (i.e., Extraversion, Agreeableness, Conscientiousness, Emotional Stability/Neuroticism, and Openness to Experience) with 12 items each, for a total of $5 \times 12 = 60$ items.

The factor loading matrix $\mathbf{\Lambda}$ described a simple measurement structure (i.e., a structure wherein every item loads on a single factor, Bollen, 1989, p. 234). The factor loadings were set to the fixed value of $\lambda = 0.85$ to represent a plausible, but reasonably high, item-scale association. We generated data with relatively high factor loadings because we wanted to mitigate the impact of measurement error on our findings without resorting to implausibly precise data. We sampled the latent scores for seven factors from a multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{\Psi}$:

$$\mathbf{\Psi} = \begin{bmatrix} 1 & \psi_{12} & \dots & \psi_{17} \\ \psi_{21} & 1 & \dots & \psi_{27} \\ \dots & \dots & \dots & \dots \\ \psi_{71} & \psi_{72} & \dots & 1 \end{bmatrix}. \tag{3.12}$$

The matrix of measurement errors $\mathbf{E}$ was sampled from a multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{\Theta}$. The off-diagonal elements of $\mathbf{\Theta}$ were set to 0 to reflect uncorrelated errors, while the diagonal elements were specified as $1 - \lambda^2$ to give the simulated items unit variances. After sampling, data were rescaled to have approximately a mean of 5 and a variance of 6.5, which are common values for Likert items in social surveys measured on a 10-point scale.

Each data matrix generated with the procedure described above was partitioned into three sub-matrices:

$$\mathbf{X} = (\mathbf{T}, \mathbf{M}, \mathbf{A}) \tag{3.13}$$

where:

- $\mathbf{T}$ is an $n \times 4$ matrix consisting of the first four indicators ($\mathbf{x}_1, \dots, \mathbf{x}_4$) of the first latent variable $\mathbf{f}_1$. We imposed missing data on these items as described in Section 3.3.1.2.

- $\mathbf{M}$ was an $n \times 4$ matrix consisting of the other four items ($\mathbf{x}_5, \dots, \mathbf{x}_8$) measuring the first latent variable. These items were used to define the probability of nonresponse for the items in $\mathbf{T}$ as described in Section 3.3.1.2.

- $\mathbf{A}$ was an $n \times 48$ matrix consisting of 48 items measuring the the remaining six latent variables $\mathbf{f}_2, \dots, \mathbf{f}_7$.

In generating the data, we varied two design factors: the number of categories into which the potential auxiliary variables were coarsened ($nCat = \infty, 7, 5, 3, 2$), and the proportion of noise variables (pn = 0, 0.33, 0.67, 1). We crossed these factors in a $5(nCat) \times 4(pn)$ factorial design. The variables in $\mathbf{M}$ and $\mathbf{A}$ represented the pool of potential auxiliary variables. We discretized these variables according to the $nCat$ factor to study the impact of data coarseness on the performance of the imputation methods. The $nCat = \infty$ level represents the uncoarsened, continuous variables. Although the data were coarsened, we applied the PCA underlying the MI-PCR methods to the Pearson correlation matrix. We recognize that we could have used polychoric, polyserial, or tetrachoric correlations, yet we purposefully chose not to do so. In the context of imputation, using alternative correlation computations impacts the imputations only through the predictive performance of PCR. Kolenikov and Angeles (2009) showed that estimating PCA based on the polychoric correlation instead of the Pearson correlation did not improve the predictive performance of PCR when applied to reduce the dimensionality of a set of ordinal predictors. Furthermore, extracting PCs based on these alternative correlations is more computationally expensive than using Pearson correlations. Considering the lack of expected advantages and the higher computational load of these alternatives, we decided to treat the ordinal data as numeric and estimate the PCA from Pearson correlations.

We used the pn factor to define the proportion of noise variables in $\mathbf{A}$. That is, the proportion of items in $\mathbf{A}$ that are uncorrelated with the items in $\mathbf{T}$. We controlled this factor at the latent variable level through the values of ψ_{jk}, the latent correlation in Equation (3.12). When $pn = 0.33$, two out of the six latent variables indicated by the items in $\mathbf{A}$ had a low correlation (0.1) with $\mathbf{f}_1$, and the remaining four had a high correlation (0.7) with $\mathbf{f}_1$. As a result, one-third of the items in $\mathbf{A}$ were also lowly correlated with the items in $\mathbf{T}$ and $\mathbf{M}$. When $pn = 0$, all latent variables were correlated at 0.7, so every variable in $\mathbf{A}$ correlated highly with the variables in $\mathbf{T}$ and $\mathbf{M}$. When $pn = 1$, all latent variables were correlated at 0.1, so all variables in $\mathbf{A}$ were trivially correlated with the variables in $\mathbf{T}$ and $\mathbf{M}$.

3.3.1.2 Missing data imposition

We imposed missing values in $\mathbf{T}$ by first generating an indicator of missingness (δ) for each column of $\mathbf{T}$. When the indicator took value 0, we left the original sampled value; when the indicator took value 1, we replaced the sampled value with a missing value. The indicator was produced by sampling from Bernoulli distributions with probabilities defined based on the following logit model:

$$logit(\delta = 1) = \beta_0 + \mathbf{M}\beta, \tag{3.14}$$

where β_0 is an intercept parameter, and β is a vector of slope parameters. Because only the variables in $\mathbf{M}$ were used to predict missingness in $\mathbf{T}$, the probability of nonresponse for a variable never depended on the variable itself. We defined the value of β_0 to align the missing values with the positive tail of $\mathbf{M}\beta$, which is a mechanism known as *right-tail MAR* (Schouten & Vink, 2021).

All slopes in β were fixed to 1, while the value of β_0 was chosen with an optimization algorithm that minimized the difference between the actual and desired proportion of missing values[14]. We fixed the proportion of missing values for each variable to 0.3, which represents a realistic—but reasonably large—value for social science data. There are no universal guidelines on which values to choose for the proportion of missing cases when conducting simulation studies. Even efforts to standardize the methodology behind missing data simulation studies struggle with providing general recommendations for this choice (Oberman & Vink, 2023). The only concrete advice is to tie the decision to the research question and reality (Graham, 2012, p. 231). We selected the proportion of missing cases to be plausible for social science data yet large enough to create substantial problems if the missing values were poorly treated[15].

Missingness was imposed using $\mathbf{M}$ in its original continuous form, even in those conditions where the potential auxiliary variables were coarsened (i.e., $nCat \neq \infty$). We made this decision to maintain the strength of the MAR mechanism as consistently as possible across conditions. Using the coarsened versions of $\mathbf{M}$ to impose missing values would have generated a weaker MAR relation (closer to missing completely at random, MCAR) for conditions with lower numbers of categories. At the same time, the solution we adopted is also imperfect. For a lower number of categories, the data available to use as predictors in the imputation models were worse representations of the actual MAR predictors. As a result, one might argue that, for the conditions with fewer categories, imputation was closer to a missing not at random (MNAR) situation rather than MAR. When designing the study we reasoned that making imputation more difficult (closer to MNAR) rather than easier (closer to MCAR) would lead to more informative results.

---

[14] The pseudo R-squared for the logistic regression of the missing value indicator on the predictors of missingness was approximately 14%. The AUC for the logistic regression was approximately 0.74.

[15] The performance of complete-case analysis can be seen in the results dashboard (Costantini, 2024d).

3.3.1.3 Imputation procedures

After generating the data and imposing missing values, every variable in $\mathbf{T}$ was imputed with the three versions of MI-PCR described above:

- MI-PCR-AUX. The PCs used in the univariate imputation model for the jth variable were estimated from the set of potential auxiliary variables (the variables in $\mathbf{M}$ and $\mathbf{A}$). For every variable under imputation, the other variables in $\mathbf{T}$ were also used as predictors.
- MI-PCR-ALL. The PCs used in the univariate imputation model for the jth variable were estimated from the entire data set ($\mathbf{T}, \mathbf{M}, \mathbf{A}$). An initial single imputation step was required to obtain a complete version of $\mathbf{T}$ from which estimate the PCs. We implemented this imputation by running a single chain of the mice algorithm for 20 iterations. We selected the predictors for this single imputation model via a correlation-thresholding strategy whereby all variables correlating at least $r = 0.3$ with the imputation targets were used as predictors.
- MI-PCR-VBV. The PCs used in the univariate imputation model for the jth variable were estimated from all other variables ($\mathbf{T}_{-j}, \mathbf{M}, \mathbf{A}$), at every iteration.

We also imputed the missing data using two non-PCR methods to act as additional points of comparison:

- MI with correlation-based thresholding (MI-QP). As a pragmatic point of comparison, this method used the *quickpred* function from the R package *mice* (van Buuren & Groothuis-Oudshoorn, 2011) to select the predictors for the univariate imputation models via the correlation-based thresholding strategy described by van Buuren et al. (1999, pp. 687–688). To implement this approach, we selected only those predictors that correlated with the imputation targets (or their associated missingness indicators) at $r = 0.1$ or higher. For every jth variable under imputation, *quickpred* selected predictors from the remaining variables ($\mathbf{T}_{-j}, \mathbf{M}, \mathbf{A}$).
- MI with oracle properties (MI-OR). This method represented the idealized, hypothetical situation wherein the imputer knows the optimal imputation model. The univariate imputation models included the remaining analysis model variables (which were also the other imputation targets, in this case) and the predictors that were used to impose missingness ($\mathbf{T}_{-j}, \mathbf{M}$). For this method, we used our perfect knowledge of the missing data mechanism to define which

variables should be predictors in the imputation models. As such, MI-OR represents an optimal point of comparison but is not replicable in practice.

To explore how the number of PCs used in MI-PCR impacts performance, we implemented the MI-PCR methods with different numbers of components. We implemented each method with fixed numbers of components (i.e., 1, 2, . . . , 10, 20, 25) as well as the maximum number of components possible (which varied by method). In PCA, the maximum number of components cannot exceed the number of rows or columns of the data, so this number depends on the specific MI-PCR implementation:

- For MI-PCR-AUX, the maximum number of PCs was $56 - 4 = 52$, the number of variables in matrices $\mathbf{M}$ and $\mathbf{A}$.
- For MI-PCR-ALL, the maximum was 56, the total number of variables in the data set.
- For MI-PCR-VBV, the maximum was $56 - 1 = 55$, the number of variables available as predictors for each univariate imputation model.

Using the maximum number of components addresses possible collinearity among the imputation model predictors without performing any dimensionality reduction.

Every imputation algorithm was used to obtain five imputed data sets, the default in the *mice* R package. All starting imputations were created by a simple random draw from the data. We set the number of iterations to 20 after checking convergence for a subset of replications. We evaluated convergence by plotting the means and standard deviations of the imputed variables. For more information on this approach, see Costantini, Lang, Reeskens, and Sijtsma (2023). Convergence plots are provided in the results dashboard (Costantini, 2024d).

3.3.1.4 Analysis and outcome measures

For each of the $S = 500$ simulated data sets, we imputed the variables in $\mathbf{T}$ with the different methods described above, and we pooled the estimates of their means, variances, covariances, and correlations[16] across the multiple imputations. The pooled estimates were stored and used to assess the performance of the imputation methods. For a given parameter ϕ (e.g., mean of $\mathbf{x}_1$, correlation between $\mathbf{x}_1$ and $\mathbf{x}_2$),

---

[16] We applied Fisher's z transformation to the correlation coefficients before pooling. We then back-transformed the pooled correlation coefficient estimates with the inverse Fisher's z transformation (van Buuren, 2018, p. 146).

we used the absolute percent relative bias (PRB) to quantify the estimation bias introduced by the imputation procedure:

$$\mathit{PRB} = \left| \frac{\bar{\hat{\phi}} - \phi}{\phi} \right| \times 100 \tag{3.15}$$

where ϕ is the true value of the focal parameter defined as $\sum_{s=1}^{S} \hat{\phi}_s^{full}/S$, with ϕ_s^{full} being the parameter estimate for the sth repetition computed on the original fully observed data. The averaged focal parameter estimate under a given missing data treatment was computed as $\bar{\hat{\phi}} = \sum_{s=1}^{S} \hat{\phi}_s/S$, with $\hat{\phi}_s$ being the estimate obtained from the treated incomplete data in the sth simulated data set. Following Muthén et al. (1987), we considered PRB > 10 as indicative of problematic estimation bias.

To measure the statistical efficiency of the imputation methods we computed the average width of the confidence intervals (CIW).

$$CIW = \frac{\sum_{s=1}^{S} \widehat{CI}_s^{upper} - \widehat{CI}_s^{lower}}{S}, \tag{3.16}$$

with $\widehat{CI}_s^{upper}$ and $\widehat{CI}_s^{lower}$ being the upper and lower bounds of the estimated confidence interval for the sth repetition. In general, narrower CIWs indicate higher efficiency. However, narrower CIWs are not preferred if they come at the expense of good confidence interval coverage (CIC) of the true parameter values. CIC is the proportion of confidence intervals that contain the true value of the parameter, across the S simulated data sets:

$$CIC = \frac{\sum_{s=1}^{S} I(\phi \in \widehat{CI}_s)}{S}, \tag{3.17}$$

where $\widehat{CI}_s$ is the confidence interval of the parameter estimate $\hat{\phi}_s$ in the sth replication, and $I(.)$ is the indicator function that returns 1 if the argument is true and 0 otherwise. CIC depends on both the bias and the CIW for a parameter estimate. An imputation method with good coverage should result in CICs greater than or equal to the nominal rate. For 95% CIs, CIC below 0.9 is usually considered problematic (e.g., van Buuren, 2018, p. 52; Collins et al., 2001, p. 340) as it implies inflated Type I error rates. High CIC (e.g., 0.99) implies inflated Type II error rates.

3.3.2 Results

3.3.2.1 Bias

In Figure 3.1, we report the PRB for the correlation coefficient between $\mathbf{x}_1$ and $\mathbf{x}_2$—two of the four imputed items in $\mathbf{T}$—in an illustrative selection of conditions. In this report, we focus on the estimates of the correlation as this was the hardest parameter to recover (i.e., the performance differences were most pronounced). The interested reader can examine the same figures for the mean, variance, and covariance in the results dashboard (Costantini, 2024d). In what follows, we write the number of components used for PCR-based methods in subscript, so that MI-PCR-AUX$_{(1)}$ refers to the use of MI-PCR-AUX with a single component. Similarly, we use the subscript to discuss the performance of PCR-based methods using a range of PCs. For example, we refer to the performance of MI-PCR-VBV using 7 to 10 PCs as MI-PCR-VBV$_{(7:10)}$.

MI-PCR-AUX resulted in acceptable bias in all conditions (PRB < 10). The bias resulting from MI-PCR-AUX depended on the number of PCs retained as predictors. The bias obtained by MI-PCR-AUX$_{(1:6)}$ was around a PRB of 2.5 for all the levels of $nCat$ and pn. MI-PCR-AUX$_{(7:10)}$ resulted in PRBs below 2.5 for $nCat = \infty$ and 5, while the bias increased to approximately 2.5 for $nCat = 2$, for both $pn = 0$ and 1.

Independently of the coarseness of the data, MI-PCR-VBV$_{(1:6)}$ resulted in PRBs between 10 and 20, and above 20, for $pn = 0$ and 1, respectively. MI-PCR-VBV$_{(7:10)}$ led to PRBs below 2.5 for $nCat = \infty$ and 5, and to PRBs around 5 for $nCat = 2$. These values were not affected by the varying proportion of noise variables.

In the condition with no noise variables, MI-PCR-ALL$_{(3)}$ already returned PRB smaller than 10 for $nCat = \infty$ and 5, while for $nCat = 2$ MI-PCR-ALL needed at least 4 components to return acceptable bias. In the conditions with $pn = 1$, the number of components needed by MI-PCR-ALL to produce PRB < 10 were 6, 5, and 4, for $nCat$ ∞, 5, and 2, respectively. As with the other MI-PCR methods, MI-PCR-ALL$_{(7:10)}$ resulted in low bias (PRB < 2.5) for both $pn = 0$ and 1. With $pn = 1$ and $nCat = 2$, MI-PCR-ALL$_{(7:10)}$ resulted in lower bias compared to other levels of $nCat$, and in the smallest bias across all methods.

MI-QP resulted in acceptable bias in all conditions (PRB < 10). The PRB obtained with this method increased as a function of the coarseness of the data: the smallest PRB was obtained for $nCat = \infty$ and the highest was obtained for $nCat = 2$. Furthermore, the bias obtained by MI-QP was smaller for $pn = 1$ than for $pn = 0$. Finally, MI-OR produced PRBs below 2.5 in all conditions and, while this performance was not affected by the proportion of noise variables, the bias slightly increased as the data were coarsened to fewer categories.

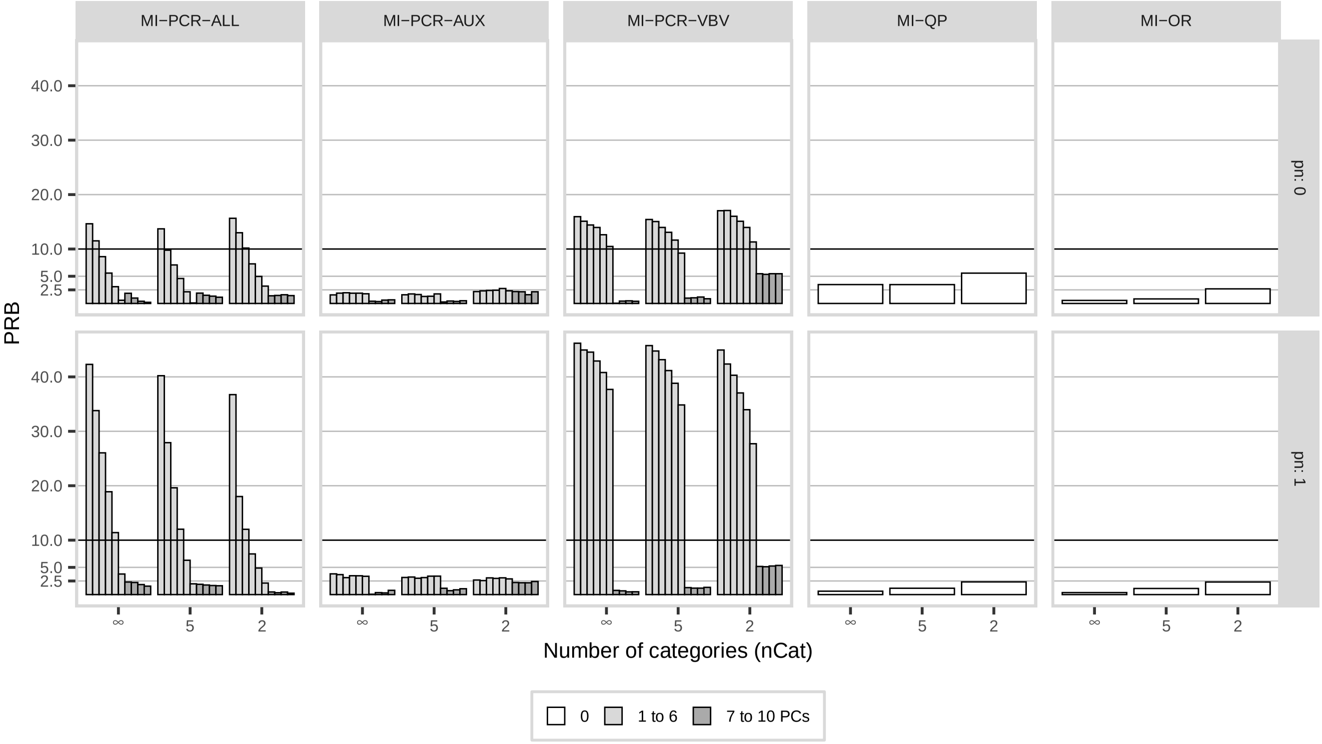


Figure 3.1: Percent relative bias for the correlation between $\mathbf{x}_1$ and $\mathbf{x}_2$ in simulation study 1. pn is the proportion of noise variables in $\mathbf{A}$. The X-axis of each histogram distinguishes three levels of coarsening for the potential auxiliary variables $(nCat = (\infty, 5, 2))$. For each MI-PCR method, we reported a different vertical bar for each PRB obtained using a different number of PCs (from 1 to 10, from left to right).

3.3.2.2 Confidence Intervals

CICs for the correlation coefficient between $\mathbf{x}_1$ and $\mathbf{x}_2$ are plotted in Figure 3.2. As a general trend, the fewer categories used for discretization, the higher the deviation from nominal coverage was. MI-PCR-ALL was the only exception to this trend, showing lower deviations of CIC from 0.95 for lower numbers of categories.

MI-PCR-AUX$_{(1:6)}$ resulted in small deviations from nominal coverage (CIC ≈ 0.9) for $pn = 0$ and in clear under-coverage (CIC < 0.9) for $pn = 1$. These trends were constant across the different levels of $nCat$. MI-PCR-AUX$_{(7:10)}$ resulted in acceptable coverage (CIC between 0.9 and 0.95) independently of pn, but smaller numbers of categories led to more deviation from nominal coverage. MI-PCR-VBV$_{(1:6)}$ led to severe under-coverage (CIC < 0.7) in all data conditions. MI-PCR-VBV$_{(7:10)}$ resulted in close to nominal coverage, with a tendency toward over-coverage (CIC > 0.95), in all conditions except the ones with $nCat = 2$, when it resulted in severe under-coverage (CIC < 0.8), for both $pn = 0$ and 1. MI-PCR-ALL$_{(1:6)}$ led to CIC < 0.9 in all data conditions, expect for MI-PCR-ALL$_{(6)}$ which produced CIC between 0.9 and 0.95 for $pn = 0$ and $nCat = \infty$ and 5. MI-PCR-ALL$_{(7:10)}$ showed more severe under-coverage than the other MI-PCR methods in all conditions except for $nCat = 2$, for both $pn = 0$ or 1. Finally, MI-QP resulted in approximately nominal coverage only in the condition with $pn = 1$ and $nCat = \infty$, and MI-OR resulted in under-coverage only for $nCat = 2$.

Figure 3.3 shows the average CIW for the correlation between $\mathbf{x}_1$ and $\mathbf{x}_2$. All MI-PCR methods using at least seven components produced narrower confidence intervals than the intervals obtained with MI-QP. MI-PCR-ALL$_{(7:10)}$ resulted in the narrowest confidence intervals, followed by MI-PCR-VBV$_{(7:10)}$, and MI-PCR-AUX$_{(7:10)}$. However, the confidence intervals obtained with MI-PCR-ALL and MI-PCR-VBV almost doubled in size when using fewer than seven components. MI-PCR-AUX$_{(1:6)}$ was less influenced by the number of PCs, providing only slightly wider confidence intervals than MI-PCR-AUX$_{(7:10)}$. MI-PCR-VBV$_{(7:10)}$ and MI-PCR-AUX$_{(7:10)}$ resulted in approximately the same CIW independently of the coarseness of the data and the proportion of noise variables. For $pn = 1$, MI-PCR-ALL$_{(1:6)}$ resulted in narrower confidence intervals when data were dichotomized compared to when they were not, while MI-PCR-ALL$_{(7:10)}$ resulted in approximately the same CIW, independently of the data coarseness.

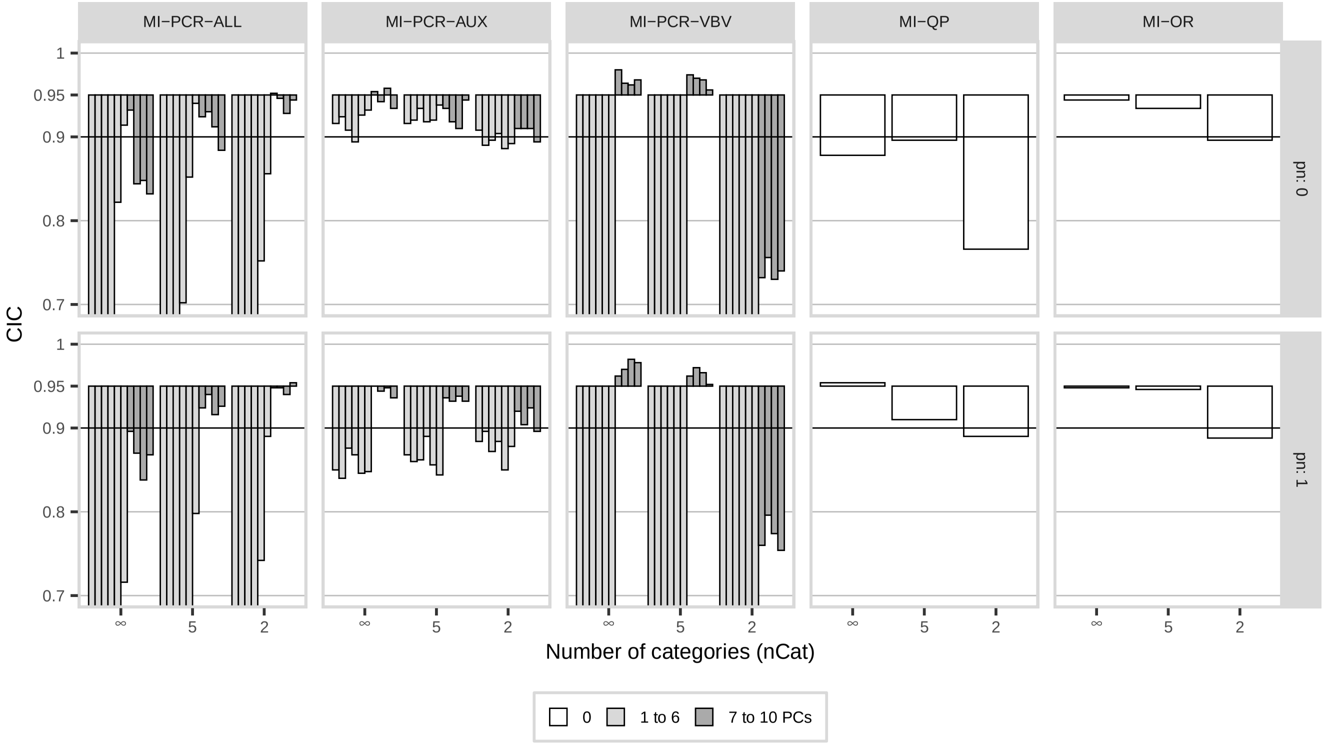


Figure 3.2: Confidence interval coverage for the correlation between $\mathbf{x}_1$ and $\mathbf{x}_2$ in simulation study 1. pn is the proportion of noise variables in $\mathbf{A}$. The X-axis of each histogram distinguishes three levels of coarsening for the potential auxiliary variables ($nCat = (\infty, 5, 2)$). For each MI-PCR method, we reported a different vertical bar for each CIC obtained using a different number of PCs (from 1 to 10, from left to right).

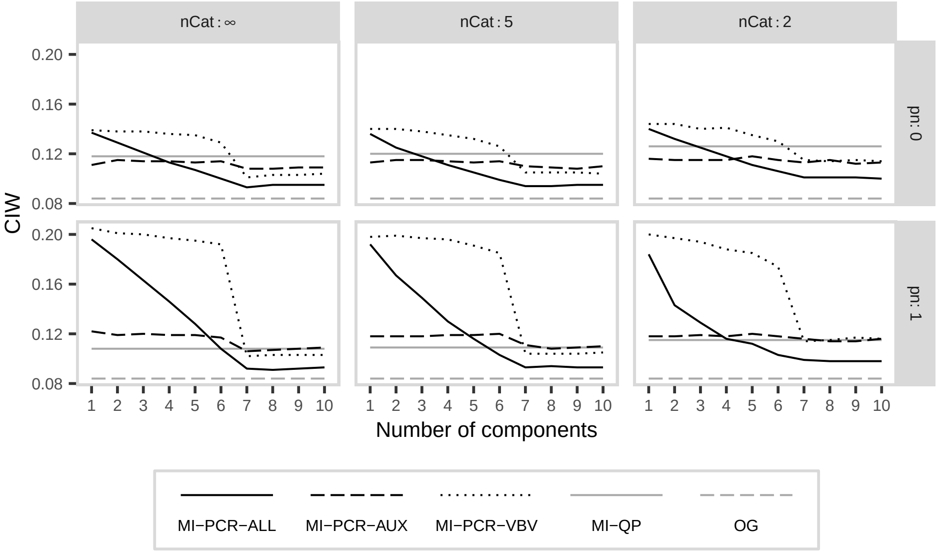


Figure 3.3: Average confidence interval width for the correlation between $\mathbf{x}_1$ and $\mathbf{x}_2$ in simulation study 1. $nCat$ is the number of categories for the items in matrices $\mathbf{M}$ and $\mathbf{A}$. pn is the proportion of noise variables in $\mathbf{A}$.

3.3.3 Discussion

The most important factor influencing the performance of the MI-PCR methods was the number of components used. This result followed a very clear, dichotomous pattern: using fewer than seven components led to poor performance across all outcome measures for all MI-PCR methods, whereas using more than seven components universally produced much better performance. As we generated the data based on seven latent variables, this result suggests that researchers using MI-PCR must employ at least as many components as there are latent variables in the data-generating model. Thankfully, except for MI-PCR-ALL, our results suggest no substantial consequences for using more components than necessary.

For all methods except MI-PCR-ALL, bias was higher when the potential auxiliary variables were discretized to fewer categories. This pattern probably reflects the loss of observed information caused by discretization. As a result of discretization, the association between variables is attenuated, which makes every auxiliary variable less useful as an imputation model predictor. Furthermore, missing values were imposed on the target variables based on the continuous variables in set $\mathbf{M}$. When the variables in set $\mathbf{M}$ were discretized, they became poorer representations of the actual MAR predictors. As a result, even MI-OR follows a trend of increasing bias for decreasing numbers of categories. The discretization of the potential auxiliary variables did not seem to impact the performance of MI-PCR more than other approaches.

MI-QP was strongly influenced by the proportion of noise variables in the set of potential auxiliary variables. The *quickpred* approach is most effective when the proportion of noise variable is high, because there is a clear distinction between variables that are correlated with the targets of imputation and those that are not. However, the method loses its efficacy as more variables correlate strongly with the imputation targets, because a large number of nearly collinear predictors end up selected into the model. For the most part, the proportion of noise variables did not have a strong impact on the performance of the MI-PCR methods. A higher proportion of noisy variables resulted in higher bias when fewer than seven components were selected. However, when enough PCs were retained, the MI-PCR methods' performances were indistinguishable across different proportions of noise variables.

The variable-by-variable approach seems to be the most promising way of using PCA within MICE. Although sometimes outperformed by MI-PCR-AUX (e.g., when using fewer than seven components or when the potential auxiliary variables were dichotomized), MI-PCR-VBV produced low bias, good coverage, and its competitive imputation performance was also accompanied by a few other desirable features. Compared to the other MI-PCR approaches, when MI-PCR-VBV deviated from nominal coverage, it showed a tendency toward over-coverage. An imputation method characterized by over-coverage will inflate type II error rates, making inferences more conservative than they should be. Although this is undesirable, it is usually less worrisome than the problem of under-coverage, which inflates type I error rates. Furthermore, MI-PCR-VBV does not rely on an initial single imputation step to obtain complete data for extracting PCs. By performing PCA at every iteration, there is no need to pre-impute the variables from which PCs are extracted. Finally, MI-PCR-VBV does not require knowledge of the analysis model, while MI-PCR-AUX needs this knowledge to distinguish which variables in the data should be summarized by PCs and which variables should be used in their raw form. At the same time, MI-PCR-VBV can still incorporate important features of the analysis model, if these features are known before imputation. Analysis model variables can be included in any desired functional form as predictors in the imputation models and excluded from the PC estimation. In such a scenario, MI-PCR-VBV would supplement each imputation model with PCs representing information that would have otherwise been ignored by the imputation procedure.

3.4 Simulation study 2: More variables

Psychological self-report inventories and large social surveys can have hundreds of variables. For example, the NEO-PR-I (Costa Jr et al., 1991) measures the same 5 personality factors as the NEO-FFI but uses 48 (instead of 12) items to define each factor. Consequently, this single personality inventory comprises 240 items. To evaluate the extent to which the results from the above simulation study generalize to problems with more variables, we replicated simulation study 1 with a larger number of potential auxiliary variables. A larger pool of potential auxiliary variables could cause problems in a couple of ways. If more auxiliary variables bring a larger number of important imputation predictors, there is an increased risk of collinearity among these predictors. Likewise, if more auxiliary variables produce more noise variables, the added noise could reduce the effectiveness of PCA as a dimensionality reduction technique (at least, with respect to the task of generating imputation model predictors).

In the second simulation study, we were only interested in exploring whether the relative performance of the methods studied was impacted by a larger dimensionality of the auxiliary set. We did not want to confound the comparison by altering the nature of the missing data problem, as well. Hence, we increased the size of the auxiliary set, $\mathbf{A}$, by increasing the number of items measuring the latent variables in $\mathbf{A}$ from 8 to 39. We kept the number of items measuring the first latent variable equal to 8 to keep the missing data problem comparable between the two simulation studies (i.e., same number of items under imputation, same number of MAR predictors, same correlation between variables with missing values and MAR predictors). As a result, the data sets we generated for the second simulation study comprised $8 + 6 \times 39 = 242$ variables. Otherwise, we used the same simulation study procedure described in 3.3.1 to generate the data, impose missing values, perform imputations, and analyze the results.

In Figures 3.4, 3.5, and 3.6, we reported the PRB, CIC, and CIW, respectively, for the correlation coefficient between $\mathbf{x}_1$ and $\mathbf{x}_2$ in an illustrative selection of conditions. The same overall patterns described in the first simulation study were still present. However, a few key differences did arise:

- In simulation study 1—with only 56 total predictors—MI-PCR-ALL and MI-PCR-VBV showed a gradual improvement in performance as more components were used. However, with $p = 242$, both methods showed a persistently high bias (PRB > 10) and low coverage (CIC < 0.7) when using fewer than 7 components. Both methods also demonstrated a more sudden improvement in performance at the 7th PC mark.

- The performance of MI-QP suffered relatively more from decreasing proportions of noise variables. In both studies, it was clear that MI-QP led to lower bias and better CIC when only a few predictors were correlated with the variables under imputation (i.e., for higher values of pn). However, in simulation study 2, the increased collinearity due to $pn = 0$, resulted in extreme bias (PRB > 20) and under-coverage of the true parameter values (CIC < 0.7).

In Figure 3.7 we reported the average imputation time in seconds for all imputation methods. MI-PCR-AUX was the fastest, taking just a few seconds to run through the five chains and 20 iterations of the mice algorithm. MI-PCR-VBV was the PCA-based method taking the longest time, with an average imputation time of around 40 seconds. MI-PCR-ALL and MI-QP were impacted by the number of noise variables in the data. Both took less than 10 seconds in the presence of many noise variables. However, for $pn = 0$, they took around 20 and 80 seconds, respectively. Both methods also took a few seconds less when the predictor data had been dichotomized.

3.5 Case study: fireworks disaster data

To understand the performance of MI-PCR in real data, we compared the performance of the three MI-PCR implementations described above to an imputation carried out by an imputation expert on a real-world data set. van Buuren (2018, pp. 315–317) gives a detailed description of how he solved the many-variables problem for the Fireworks' disaster data set (FDD). On May 13, 2000, the explosion of a fireworks storage facility in Enschede, the Netherlands, killed 23 people and injured approximately 950 others. Many people residing in the neighborhood of the explosion experienced signs of post-traumatic stress disorder (PTSD). The FDD was collected as part of a randomized controlled trial carried out in the aftermath of the explosion. The data were collected to assess the efficacy of two treatments of anxiety-related disorders in children, in terms of reducing PTSD symptoms over time. The main outcome measure for this analysis was the PTSD Reaction Index (PTSD-RI), measured as reported by the child and by the parent, at three different time points. Fifty-two children were assigned either Eye Movement Desensitisation and Reprocessing (EMDR) treatment or Cognitive Behavioral Therapy (CBT). Of the 65 variables recorded in the data set, 49 were incomplete. The percentage of missing values on each variable ranged from 3% to 50%. A complete-case analysis would

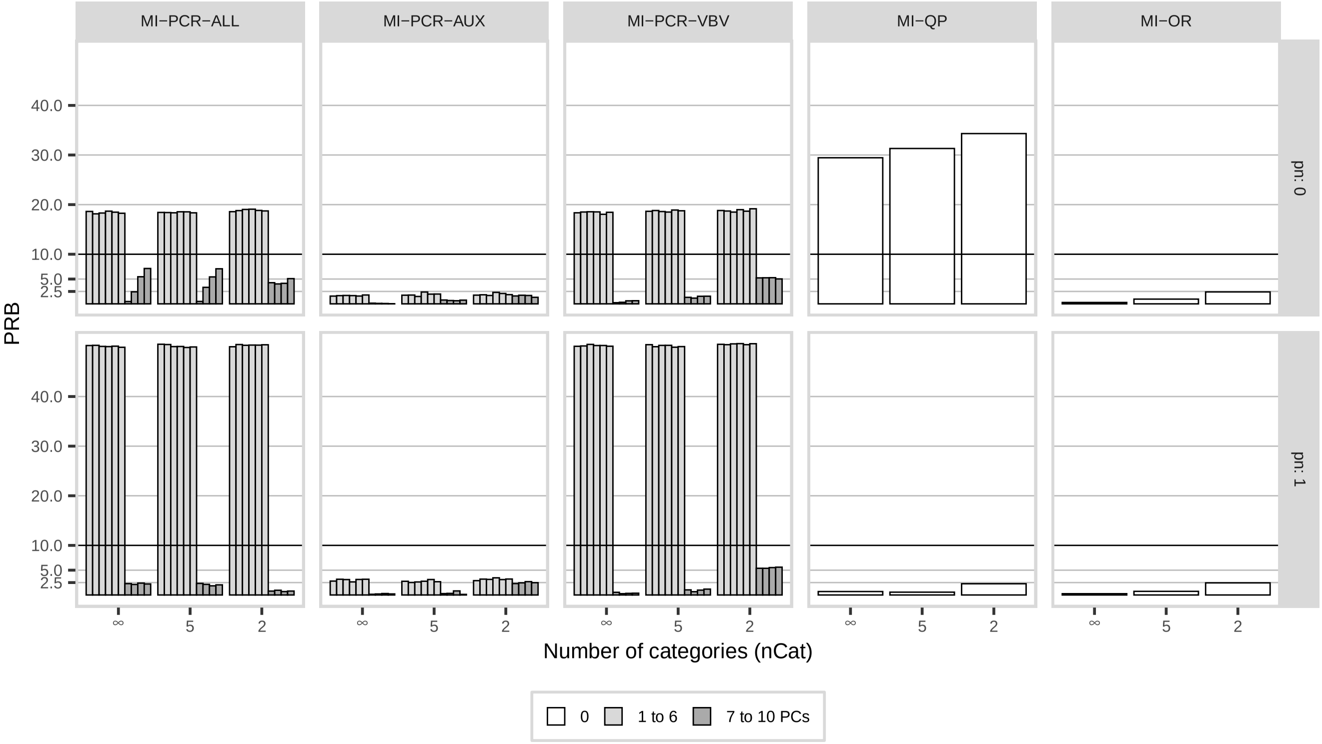


Figure 3.4: Percent relative bias for the correlation between $\mathbf{x}_1$ and $\mathbf{x}_2$ in simulation study 2. pn is the proportion of noise variables in $\mathbf{A}$. The X-axis of each histogram distinguishes three levels of coarsening for the potential auxiliary variables ($nCat = (\infty, 5, 2)$). For each MI-PCR method, we reported a different vertical bar for each PRB obtained using a different number of PCs (from 1 to 10, from left to right).

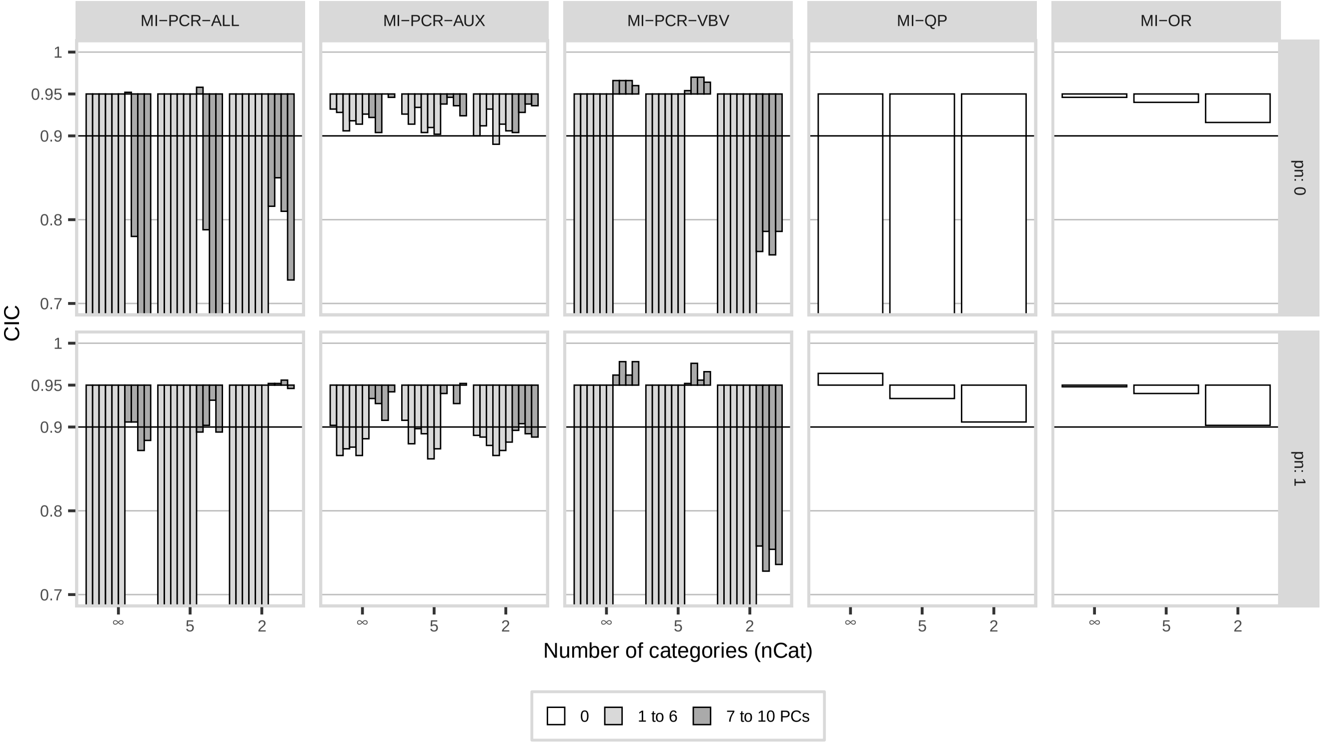


Figure 3.5: Confidence interval coverage for the correlation between $\mathbf{x}_1$ and $\mathbf{x}_2$ in simulation study 2. pn is the proportion of noise variables in $\mathbf{A}$. The X-axis of each histogram distinguishes three levels of coarsening for the potential auxiliary variables ($nCat = (\infty, 5, 2)$). For each MI-PCR method, we reported a different vertical bar for each CIC obtained using a different number of PCs (from 1 to 10, from left to right).

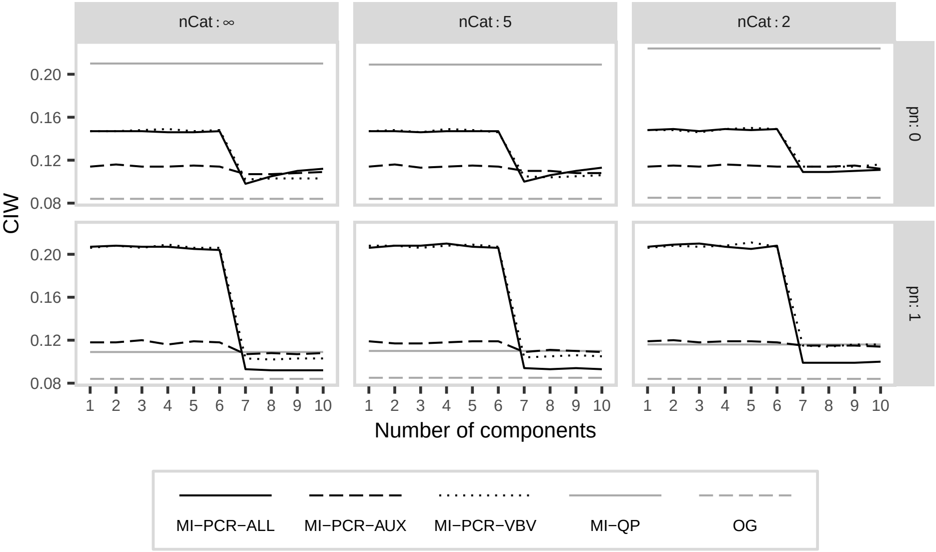


Figure 3.6: Average confidence interval width for the correlation between $\mathbf{x}_1$ and $\mathbf{x}_2$ in simulation study 2. $nCat$ is the number of categories for the items in matrices $\mathbf{M}$ and $\mathbf{A}$. pn is the proportion of noise variables in $\mathbf{A}$.

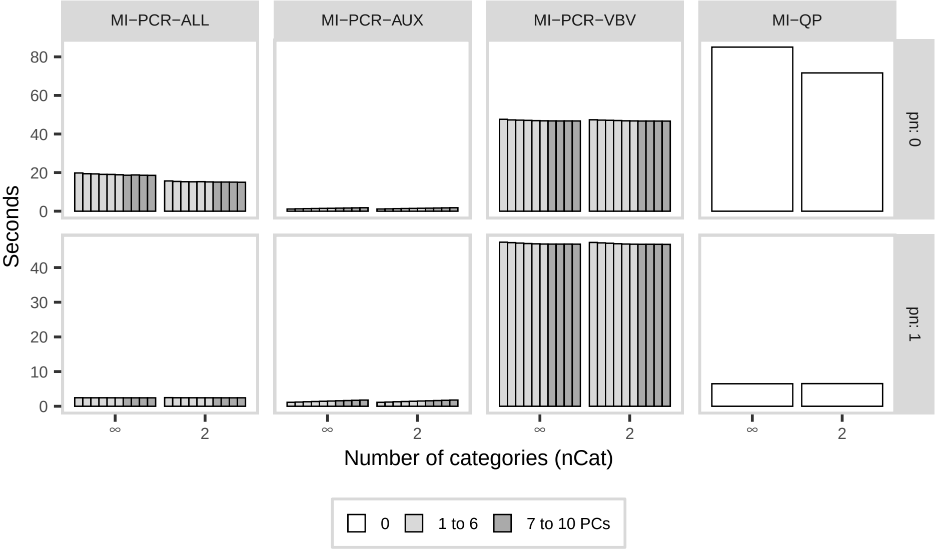


Figure 3.7: Average imputation time in simulation study 2. $nCat$ is the number of categories for the items in $\mathbf{M}$ and $\mathbf{A}$. pn is the proportion of noise variables in $\mathbf{A}$. For each MI-PCR method, we reported a different vertical bar for the average imputation time obtained using a different number of PCs (from 1 to 10, from left to right).

have resulted in analyzing only eight cases. To avoid the unacceptable reduction in sample size and biased parameter estimates, a principled missing data treatment was needed.

The major difficulty in imputing these data was the large number of predictors relative to the sample size ($p = 65, n = 52$). To avoid over-parameterized imputation models, the imputation models' predictors needed to be carefully selected. Van Buuren employed two main strategies to address the high-dimensional nature of the data:

- Use only the first measurement of the outcomes as predictors in the imputation models of other outcomes. This choice reduced the number of predictors by two-thirds.
- Use the total scores of scales as predictors in the univariate imputation models of other variables, instead of the individual scale items. This was done using passive imputation (van Buuren and Oudshoorn, 2000, p. 12; Eekhout et al., 2018, pp. 1129–1130).

Each of the three MI-PCR strategies considered in this report is suitable to address the large number of potential imputation model predictors in the FDD. The advantage of MI-PCR is the automatic way in which large numbers of predictors can be accommodated. Through this case study, we wish to illustrate the degree to which MI-PCR can produce results similar to those obtained by an expertly designed imputation procedure. There are several reasons why the FDD is ideally suited to this purpose:

- The data have key characteristics of social and behavioral science data sets (e.g., composite scales, longitudinal data).
- The code van Buuren used to perform the MI procedure is freely accessible.[17]
- The data are freely accessible through the *mice* R package (van Buuren & Groothuis-Oudshoorn, 2011).
- The reasoning behind the expert's MI procedure is well documented (van Buuren, 2018, p. 313).

---

[17] https://github.com/stefvanbuuren/fimdbook/blob/master/R/fimd.R

3.5.1 Method

The main analysis reported in van Buuren (2018, p. 313) focused on the effect of treatment on the mean PTSD-RI scores (both child-reported and parent-reported) across three time points. Therefore, six variables were analyzed: the PTSD-RI total scores reported by children and their parents at three different time points. We imputed these six analysis variables with the three MI-PCR procedures evaluated above and compared the pooled means obtained thereby with the results of van Buuren's imputation procedure. To evaluate the variability of the imputation methods, we repeated this procedure with 20 different random number seeds.

The results of the simulation study suggested that MI-PCR performs poorly if an insufficient number of components is extracted, whereas its performance is not severely impacted by selecting too many components. Therefore, to choose the number of PCs in this case study, we decided to use the maximum number of PCs allowed by each imputation procedure. The two variables with the most missing values had 25 cases observed. Hence, at most 24 predictors could be used for each imputation model, and we could use at most 24 PCs.

We specified MI-PCR-AUX to impute the six items under analysis. In each of the six univariate imputation models, MI-PCR-AUX used as predictors the other five variables under imputation and the first 19 PCs estimated from the remaining auxiliary columns (5 + 19 = 24). We performed single imputation on the potential auxiliary variables to allow the PC estimation. We used predictive mean matching as univariate imputation method and kept the imputations obtained after 20 iterations. The predictors for the imputation models were selected by correlation-thresholding, with the threshold set to 0.1.

The MI-PCR-ALL method used the first 24 PCs estimated on the entire data set—including the 6 analysis variables—as the sole predictors in each of the six univariate imputation models. The same single imputation specification used for MI-PCR-AUX was used here to generate the complete data from which to estimate the PCs. Finally, MI-PCR-VBV was performed by extracting 24 components from all the variables not under imputation for each univariate imputation model, at every iteration of the mice algorithm.

All starting imputations were created by a simple random draw from the data. Convergence of the pre-processing single imputation used for MI-PCR-AUX and MI-PCR-ALL was assessed with the trace plots of the imputed values' means and standard deviations produced by the `plot.mids()` function from the *mice* package. We assessed the convergence of all the imputation methods with the same technique.

We considered all methods to have converged within 20 iterations. The plots of convergence trends can be viewed in the results dashboard (Costantini, 2024d).

3.5.2 Results

Figures 3.8 and 3.9 report the (pooled) mean level of PTSD-RI over the three time points after imputation with the following approaches:

- Expert's imputation models;
- The three MI-PCR approaches;
- Default run of *mice* without any pre-processing and using all default argument values.

While all methods led to similar trends, variability of the imputations was noticeably higher for MI-PCR-AUX, MI-PCR-ALL, and the default run of *mice*. At every time point, the 20 different pooled means of the analyzed variables spread over a wider range of values compared to the expert's imputation. This pattern held for both outcome variables, but it was most conspicuous for the child-reported PTSD-RI. MI-PCR-AUX and MI-PCR-ALL had lower imputation precision than even the default run of *mice* in this setup. The performance of MI-PCR-VBV was on par with that of the expert's imputation: both methods produced comparable location and spread of the outcome variables pooled means.

3.6 General discussion

In this study, we were interested in understanding the performance of the various MI-PCR methods as a function of the number of components used, the coarseness of the predictor data, and the amount of noise in the data. Our findings suggest that MI-PCR performs well across a wide range of conditions, and MI-PCR-VBV shows the best performance of the three implementations we evaluated. So long as the number of PCs met or exceeded the true number of latent variables, MI-PCR-VBV outperformed the standard correlation-thresholding approach (MI-QP) on all the metrics we considered. Furthermore, the good imputation performance of MI-PCR-VBV comes with some desirable features that are missing from the other MI-PCR methods. First, MI-PCR-VBV does not rely on knowledge of the analysis model, although such knowledge is easily incorporated when available. Second, MI-PCR-VBV does not require a pre-processing single imputation step. Third, when

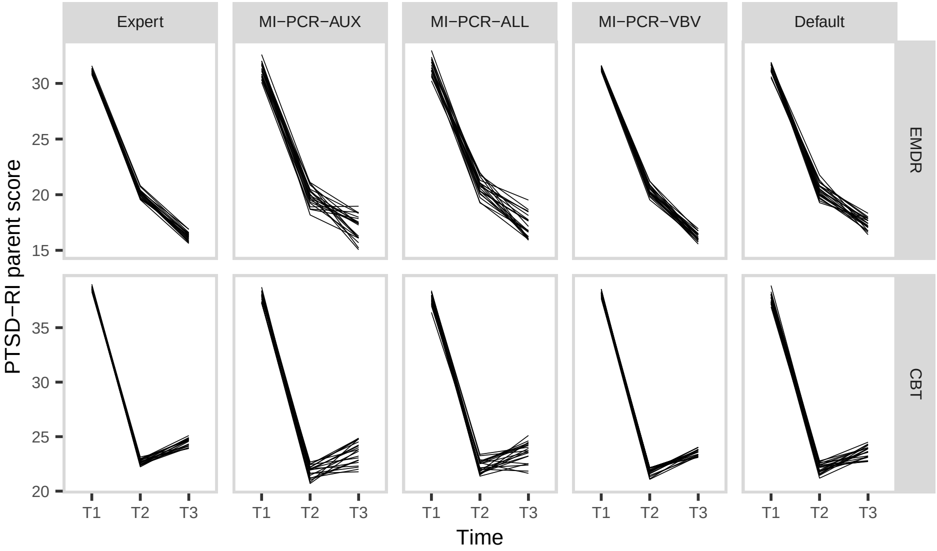


Figure 3.8: Mean levels of PTSD-RI parent score after imputation. The multiple lines plotted for each method represent results obtained with 20 different seeds.

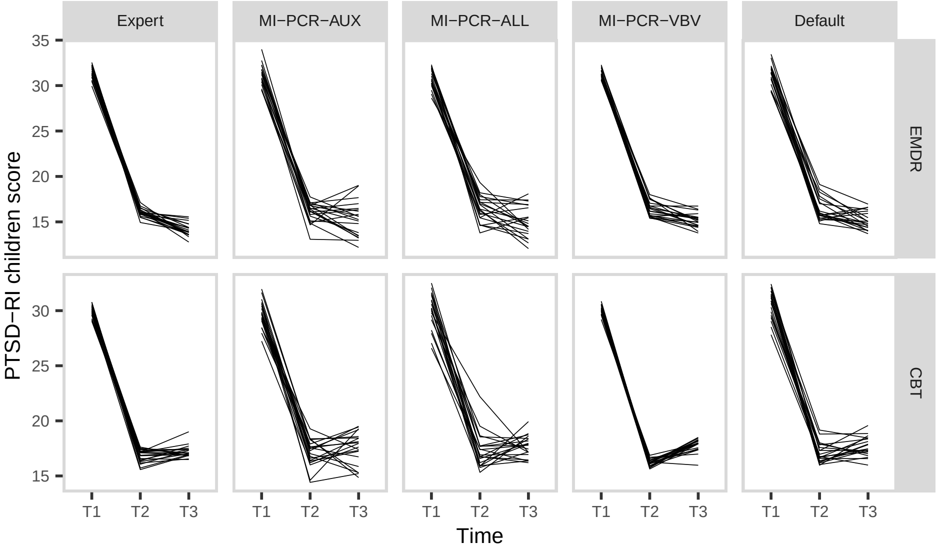


Figure 3.9: Mean levels of PTSD-RI children score after imputation. The multiple lines plotted for each method represent results obtained with 20 different seeds.

MI-PCR-VBV resulted in deviations from nominal coverage, it tended towards over-coverage. Finally, in the case study, MI-PCR-VBV was able to automatically obtain results that were essentially indistinguishable from those produced by an expertly designed imputation model.

The good performance of MI-PCR-VBV comes at the expense of computation time. Performing PCA for every variable under imputation at every iteration of the MICE algorithm requires a much larger number of computations than the other two MI-PCR methods which leads to a drastically higher imputation time. As the number of variables to impute increases, computation time might become excessive. This makes the MI-PCR-VBV strategy more suitable for broad and intermediate imputation scopes (van Buuren, 2018, p. 46), where imputation is performed once by an institution with adequate computational resources, and then delivered to a collection of researchers to be used in their analysis.

3.7 Some final remarks on the number of components

In practice, imputers using MI-PCR need a decision rule to select the number of components. Using the same number of components as the total available variables, as we did in Section 3.5, is not a viable solution for data sets with hundreds of variables (or more). Fortunately, a variety of decision rules have been proposed for this purpose (Zwick & Velicer, 1986). Based on the results described in Sections 3.3 and 3.4, we can infer that any decision rule that selects the “true” number of components, or more, would produce satisfactory results with MI-PCA.

To gain some preliminary insight into how these decision rules could impact the performance of MI-PCA, we applied four non-graphical decision rules[18] described by Raíche et al. (2013) to 500 data sets generated according to the procedure described in Sections 3.3 and 3.4. Specifically, we implemented the optimal coordinates index (*oc*), the acceleration factor (*af*), the Kaiser criterion (*kc*), and the parallel analysis criterion (*pa*) using both the fully observed, discretized data and the complete cases available after imposing the missing values.

In Table 3.1, we report the lowest, the highest, and the median number of principal components retained with each decision rule across the 500 data sets. For each decision rule, the number of PCs selected when analyzing the complete cases was always equal to or higher than the number of PCs selected when analyzing the fully observed data. The *oc* criterion demonstrated mixed performance. The median

---

[18] We used the implementation of these rules provided by the ‘nScree()’ function in the R package *nFactor* (Raíche, 2010).

number of PCs selected by *oc* was always at least 7, but the minimum number of components selected was less than 7 in all conditions. The *kc* and *pa* decision rules selected between 7 and 15 PCs when applied to data sets with $P = 56$ columns and between 7 and 39 PCs when applied to data with $P = 242$ columns. Hence, *kc* and *pa* appear to be safe options when considering our desideratum of selecting no fewer than the true number of components. In line with results presented by Raíche et al. (2013), *af* underestimated the number of PCs to retain and always selected fewer than 7 PCs. Based on these results, we can tentatively suggest applying the Kaiser criterion or the parallel analysis criterion to the complete case as a viable method of selecting the number of PCs to use in MI-PCA.

	data	Fully observed												Complete cases											
	pn	0				0.67				1				0				0.67				1			
	rule	oc	af	kc	pa	oc	af	kc	pa	oc	af	kc	pa	oc	af	kc	pa	oc	af	kc	pa	oc	af	kc	pa
P = 56																									
	highest	7	**1**	7	7	7	**1**	7	7	7	7	7	7	7	**1**	7	7	7	7	7	7	7	7	7	7
nCat = ∞	median	7	**1**	7	7	7	**1**	7	7	7	7	7	7	7	**1**	7	7	7	**1**	7	7	7	7	7	7
	lowest	**1**	**1**	7	7	**1**	**1**	7	7	**1**	**1**	7	7	**1**	**1**	7	7	**2**	**1**	7	7	**1**	**1**	7	7
	highest	7	**1**	7	7	7	**1**	7	7	7	7	7	7	8	**1**	8	8	8	7	8	8	7	7	7	7
nCat = 5	median	7	**1**	7	7	7	**1**	7	7	7	7	7	7	7	**1**	7	7	7	**1**	7	7	7	7	7	7
	lowest	**1**	**1**	7	7	**2**	**1**	7	7	**1**	**1**	7	7	**1**	**1**	7	7	**1**	**1**	7	7	**1**	**1**	7	7
	highest	8	**1**	8	8	8	**1**	8	8	8	7	8	8	14	**1**	15	15	13	7	13	13	13	7	13	13
nCat = 2	median	7	**1**	7	7	7	**1**	7	7	7	7	7	7	11	**1**	12	12	10	**1**	11	11	10	7	10	10
	lowest	**1**	**1**	7	7	**1**	**1**	7	7	**1**	**1**	7	7	**1**	**1**	9	9	**1**	**1**	8	8	**1**	**1**	8	8
P = 242																									
	highest	7	**1**	7	7	7	**5**	7	7	7	**6**	7	7	35	**1**	35	35	28	**6**	29	29	25	**6**	25	25
nCat = ∞	median	7	**1**	7	7	7	**1**	7	7	7	**1**	7	7	21	**1**	25	25	19	**1**	21	21	18	**6**	19	19
	lowest	**1**	**1**	7	7	**2**	**1**	7	7	**1**	**1**	7	7	**2**	**1**	14	14	**2**	**1**	14	14	**1**	**1**	14	14
	highest	7	**1**	7	7	7	**5**	7	7	7	**6**	7	7	39	**1**	39	39	32	**6**	32	32	27	**6**	31	31
nCat = 5	median	7	**1**	7	7	7	**1**	7	7	7	**1**	7	7	23	**1**	28	28	22	**1**	24	24	20	**6**	22	22
	lowest	**1**	**1**	7	7	**2**	**1**	7	7	**1**	**1**	7	7	**3**	**1**	19	19	**3**	**1**	17	17	**3**	**1**	16	16
	highest	13	**1**	13	13	13	**5**	13	13	13	**6**	13	13	41	**1**	41	41	35	**6**	39	39	34	**6**	34	34
nCat = 2	median	10	**1**	10	10	10	**5**	10	10	10	**1**	10	10	29	**1**	34	34	28	**5**	30	30	26	**6**	29	29
	lowest	**1**	**1**	8	8	**2**	**1**	8	8	**1**	**1**	8	8	**1**	**1**	26	26	**1**	**1**	24	24	**3**	**1**	22	22

Table 3.1: The lowest, the highest, and the median number of principal components selected by four non-graphical decision rules across 500 data sets generated according to the simulation design described in Sections 3.3 and 3.4. The decision rules reported are the optimal coordinates index (oc), the acceleration factor (af) the Kaiser criterion (kc), and the parallel analysis criterion (pa). The table distinguishes between the results obtained when applying the four decision rules to two types of data (the originally fully observed data and the complete cases), data with two different dimensionalities ($P = 56, 242$), data with different discretization levels ($nCat = \infty, 5, 2$), and data with different proportions of noise variables ($pn = 0, 0.67, 1$). Numbers below 7 are reported in bold.

3.8 Limitations and future directions

In the preceding section, we provide some preliminary insight into how four non-graphical PC enumeration methods might affect the performance of MI-PCR, but our results support only tentative recommendations. Future research should focus specifically on the issue of PC enumeration in MI-PCA and thoroughly explore which rule is most suitable for the imputation task. Furthermore, the unsupervised nature of PCA introduces an additional dimension into the decision calculus. The MI-PCR implementations we compared here extract PCs without considering the relationship between the imputation model's predictors and outcome. Consequently, the MI-PCR methods we evaluated could, potentially, extract components that explain relatively little variance in the variables under imputation, regardless of how many PCs are retained. To mitigate this possibility, MI-PCR-VBV could be implemented with some form of supervision, such as supervised PCA (Bair et al., 2006) or principal covariates regression (de Jong & Kiers, 1992). In addition to avoiding poorly predictive PCs, a supervised version of MI-PCR might require fewer components to obtain the same imputation quality. We are currently exploring these possibilities in a follow-up to the study reported here.

It is common for social scientists to analyze non-normal data such as ordinal rating scales (e.g., any item in the NEO-FFI and NEO-PI-R) or skewed social variables (e.g., items affected by extreme response styles). The results of this study apply directly to situations wherein the non-normal variables are the targets of imputation. MI-PCR only adjusts the right-hand-side of the univariate imputation models. So, our approach can be directly applied to univariate imputation models for non-normal data (e.g., Bayesian logistic and polytomous regression models, predictive mean matching). However, using PCA to extract components from a set of non-normal predictors requires more careful consideration. In general, as a tool to summarize variation on a set of variables, PCA does not need to meet rigorous distributional assumptions (Jolliffe, 2002, pp. 19, 49, 338). However, when the variables in $\mathbf{X}$ deviate from normality because of the presence of extreme cases, robust PCA can be used to reduce the impact these observations have on the estimation of the PCs. For example, Croux and Ruiz-Gazen (2005) and Hubert et al. (2009) proposed alternative PCA computations that are robust to outliers and asymmetry. Similarly, PCAMIX (Chavent et al., 2012; Kiers, 1991) can be used in the presence of a mix of continuous, ordinal, and nominal variables. Luckily, MI-PCR has a modular structure, and the classical PCA estimation we applied in this study can be replaced by any alternative PCA approach. These alternatives could improve the performance of MI-

PCR when it is applied to non-normal data, but we do not expect this change to have an impact on the relative performances of the different MI-PCR implementations we evaluated here. Any possible improvement in the quality of the PCs would impact each implementation equally, so our overall conclusions would be unlikely to change. Nevertheless, it would be interesting to evaluate the extent of the improvement that could be achieved by incorporating more robust versions of PCA.

The missing data mechanism studied in this paper is relatively simple. The probability of missing values on $\mathbf{T}$ depends, through a logit function, on the linear combination of the predictors $\mathbf{M}$. However, interactions and polynomial terms might be present in the linear component of Equation (3.14). MI-PCR can address this complexity by augmenting the set of variables from which PCs are extracted with all the interaction and polynomial terms of interest. To what extent this strategy is feasible and effective remains to be explored. Furthermore, while we focused on right tail MAR, Schouten and Vink (2021) and Collins et al. (2001) have shown that the “shape” of the missing data mechanism has an impact on the severity of a missing data problem. In the worst case, we expect the different MAR shapes to impact the absolute performance of the MI-PCR methods compared in this study, but not their relative performance. Future research could address this issue in detail.

Multilevel data provide a similar avenue for future research. Social science data are often characterized by clusters of observations. Imputation procedures that ignore this feature of the data can lead to biased estimates as imputations are generated without considering cluster dependencies (Reiter et al., 2006). One way to address this issue is to include dummy variables representing the cluster effects in the imputation models (fixed effect imputation, Enders et al., 2016). However, this approach has the disadvantage of increasing the dimensionality of the design matrix to an impractical extent. Using PCs to reduce dimensionality might be a good way to address grouping in the data without incurring estimation difficulties of sophisticated multilevel imputation procedures.

Finally, the good performance of MI-PCR-ALL in the conditions with dichotomized auxiliary variables remains something of a mystery. We did not expect this pattern, and we do not have an explanation for this finding. Since the other MI-PCR methods performed at their worst in the $nCat = 2$ condition, it would be interesting to further explore the capabilities of MI-PCR-ALL in this special case.

3.9 Conclusions

This study extends and refines the findings of Costantini, Lang, Reeskens, and Sijtsma (2023) by providing further information on how to best incorporate PCA into MI. In our simulation studies, using PCR as a univariate imputation method within every iteration of a MICE algorithm (i.e., MI-PCR-VBV) provided small bias, good statistical efficiency, and close to nominal coverage. Our case study added to these findings by showing that MI-PCR-VBV can provide performance on-par with expertly designed imputation. Although computational demand could become a limiting factor in some situations, our findings suggest that MI-PCR-VBV is a promising general-purpose, imputation algorithm that can streamline the process of conducting principled MI in data sets with many variables.

Chapter 4

Supervised dimensionality reduction for multivariate imputation by chained equations

Abstract Multivariate imputation by chained equations (MICE) is one of the most popular approaches to address missing values in a data set. This approach requires specifying a univariate imputation model for every variable under imputation. The specification of which predictors should be included in these univariate imputation models can be a daunting task. Principal component analysis (PCA) can simplify this process by replacing all of the potential imputation model predictors with a few components summarizing their variance. In this article, we extend the use of PCA with MICE to include a supervised aspect whereby information from the variables under imputation is incorporated into the principal component estimation. We conducted an extensive simulation study to assess the statistical properties of MICE with different versions of supervised dimensionality reduction and we compared them with the use of classical unsupervised PCA as a simpler dimensionality reduction technique.

Results dashboard To run the results dashboard accompanying this chapter install the Shiny app as an R package from the Zenodo permanent repository:

```
# Install shiny app
devtools::install_url(
        "https://zenodo.org/records/10759119/files/EdoardoCostantini/plotmispcr-v2.2.zip"
)
```

Then, you can start the app by running this command:

```
# Start the app
plotmispcr::start_app()
```

Reproducibility statement The R code used to produce the results reported in this chapter can be found in the permanent repository stored on Zenodo (Costantini, 2023e). The README.md file contains instructions on how to reproduce the results.

Costantini, E., Lang, K. M., & Sijtsma, K. (2023). Supervised dimensionality reduction for multiple imputation by chained equations. *arXiv preprint*. https://doi.org/10.48550/arXiv.2309.01608

4.1 Introduction

Multiple Imputation (MI) is a state-of-the-art missing data treatment in today's data analysis (Schafer and Graham, 2002; van Buuren, 2018, p. 30). Multiple imputations is often implemented through the multivariate imputation by chained equations approach (MICE, van Buuren & Oudshoorn, 2000), which can accommodate a wide range of data measurement levels. This flexibility comes from the possibility of modeling the multivariate joint density of the variables with missing values through a collection of conditional densities for every variable under imputation.

MICE requires specifying a different univariate imputation model for each variable under imputation, which entails deciding on the imputation model form and which predictors to use. The first decision is usually guided by the measurement level of the variables under imputation. For example, continuous variables can be imputed using a linear regression model, while binary variables can be imputed using logistic regression. The second decision concerns which and how many predictors to include in the imputation models and therefore it is more difficult. The general recommendation has been to follow an inclusive strategy (Collins et al., 2001), meaning that as many predictors as possible should be included in the imputation models. Using as much information as possible from the data leads to multiple imputations that have minimal bias and maximal efficiency (Collins et al., 2001; Meng, 1994). Furthermore, including more predictors in the imputation models makes the missing at random assumption (MAR) more plausible (Collins et al., 2001, p. 339). Finally, if the imputation model omits variables that are part of the analysis model fitted to the data after imputation, the parameter estimates might be biased (Enders, 2010, p. 229) and estimated confidence intervals might be too wide (R. J. A. Little & Rubin, 2002, p. 218). As a result, including more predictors in the imputation models increases the range of analysis models that can be estimated with a given set of imputations (Meng, 1994).

Despite its advantages, the inclusive strategy easily results in singularity issues (Hastie et al., 2009, p. 46) when estimating imputation models. Consequently, researchers performing MI often face difficult choices on how many and which variables to use as predictors. High-dimensional prediction models offer an opportunity to specify the imputation models automatically and to include more predictors than traditionally possible. For example, ridge regression (Hoerl & Kennard, 1970) can estimate regression models with hundreds of predictors; lasso regression (Tibshirani, 1996) can perform data-driven variable selection; decision trees (e.g., Breiman, 2001) can consider hundreds of variables for their splitting rules; and principal component analysis (PCA, Jolliffe, 2002) can summarize a large set of predictors using a few independent linear combinations.

All of these modeling strategies have been implemented in combination with MICE (Burgette & Reiter, 2010; Deng et al., 2016; Doove et al., 2014; Howard et al., 2015; Shah et al., 2014; Zhao & Long, 2016). Costantini, Lang, Reeskens, and Sijtsma (2023) compared their performance in terms of estimation bias and confidence interval coverage when applied to data with missing values. They found that using PCA to create summaries of the many possible imputation model predictors performs particularly well. In a follow-up study, Costantini et al. (2024) explored different ways of using PCA with the MICE algorithm and found that updating the principal components (PCs) for the imputation of every variable at every iteration provided the lowest bias and the lowest deviation from nominal confidence interval coverage. However, these results relied heavily on the number of components computed. With their simulation study, Costantini et al. (2024) showed that to achieve small bias and satisfactory coverage, a researcher imputing the data using PCA to aid imputation model specification should retain at least as many PCs as the number of latent variables in the data-generating model. This is undesirable as researchers usually do not know the true number of latent variables.

PCA is an unsupervised dimensionality reduction technique that summarizes the variability of a set of P variables $\{\mathbf{x}_1, \dots, \mathbf{x}_p\}$ measured on N observations with a set of Q PCs, with $Q < P$. The PCs can be used in any regression model as a replacement for the original predictors, an approach known as Principal Component Regression (PCR; Jolliffe, 2002, pp. 168-173). PCR addresses possible multicollinearity issues afflicting the model. However, the PCs obtained by PCR cannot take into account variables that are not part of the set $\{\mathbf{x}_1, \dots, \mathbf{x}_p\}$, a feature that can result in PCs that are unrelated or only weakly related to the dependent variable, which by definition is not included in the set of predictors. Contrary to PCA, supervised dimensionality reduction (SDR) techniques use the outcome variable to guide the computation so that the resulting PCs are both good representations of the predictor variables and strongly associated with the dependent variable (e.g., Bair et al., 2006; de Jong & Kiers, 1992; Wold, 1975). Using SDR within MICE might relax the need to know the number of latent variables in the data-generating model described by Costantini et al. (2024) for PCA. The purpose of this study is to evaluate how SDR techniques can improve upon unsupervised PCR as a univariate imputation model in MICE.

In this study, we considered two questions. First, what are the statistical properties (bias, coverage, confidence interval width) of parameters estimated from data treated with the MICE algorithm using different versions of SDR as the univariate imputation models? Second, can using SDR in MICE relax the PCA requirement of using at least as many PCs as the number of latent variables in the data-generating model? We used a Monte Carlo simulation study to explore the performance of different versions of SDR with the MICE algorithm.

The article is structured as follows. In Section 4.2, we describe the MICE algorithm, unsupervised and supervised dimensionality reduction, different versions of SDR, and we propose uses of SDR as a univariate imputation method for MICE. In Section 4.3, we describe the Monte Carlo simulation study. Next, we discuss the main findings (Section 4.4), our ideas for future research directions (Section 4.5), and we provide concluding remarks (Section 4.6).

4.2 Imputation methods and algorithms

We use the following notation. Indices and scalars are denoted by lowercase and uppercase letters. For example, i is an index enumerating iterations out of I total iterations ($i \in \{1, \dots, I\}$). Vectors are written in bold lowercase while matrices are denoted by bold uppercase letters. The superscript $'$ defines the transpose of a matrix. We use the subscript 'obs' and 'mis' to refer to the observed and missing elements in a vector or matrix.

4.2.1 Multivariate imputation by chained equations

Consider data set $\mathbf{Z}$ with N rows and P columns $\mathbf{z}_1, \dots, \mathbf{z}_P$. We assume that $\mathbf{Z}$ is a random draw from a multivariate distribution $f(\mathbf{Z}|\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector of unknown parameters that completely specifies its multivariate distribution. Let the first J columns of $\mathbf{Z}$ have missing values. MICE is an iterative algorithm for imputing multivariate missing data on a variable-by-variable basis. It obtains multiple imputations for the missing values by drawing from the variable-specific conditional distributions of the form:

$$f(\mathbf{z}_j|\mathbf{Z}_{-j}, \boldsymbol{\theta}_j), \tag{4.1}$$

where $\mathbf{z}_j$ is a partially observed variable, $\mathbf{Z}_{-j}$ is the collection of variables in $\mathbf{Z}$ excluding $\mathbf{z}_j$, and $\boldsymbol{\theta}_j$ is a vector of model parameters. The parameters $\boldsymbol{\theta}_j$ are specific to the respective conditional distributions and might not determine the unique *true* joint distribution $f(\mathbf{Z}|\boldsymbol{\theta})$. We refer to the conditional distributions in Equation (4.1) as (univariate) imputation models.

The MICE algorithm starts by replacing the missing values in each $\mathbf{z}_j$ with initial guesses (e.g., random draws from the observed values). At iteration i, the MICE algorithm imputes successively variables $\mathbf{z}_1$ to $\mathbf{z}_J$ by taking draws from the following distributions:

$$\boldsymbol{\theta}_j^{(i)} \sim f(\boldsymbol{\theta}_j | \mathbf{Z}_{j,\text{obs}}, \mathbf{Z}_{-j}^{(i)}), \tag{4.2}$$

$$\mathbf{z}_{j,\text{mis}}^{(i)} \sim f(\mathbf{z}_{j,\text{mis}} | \mathbf{Z}_{-j}^{(i)}, \boldsymbol{\theta}_j^{(i)}) \tag{4.3}$$

Equation (4.2) is the fully conditional posterior distribution defined by the product of an uninformative prior distribution for $\boldsymbol{\theta}_j$ and the likelihood of observing $\mathbf{z}_{j,\text{obs}}$ under the imputation model for $\mathbf{z}_j$. Equation (4.3) is the posterior predictive distribution from which updates of the imputations are drawn. In both equations, $\mathbf{Z}_{-j}^{(i)}$ is $(\mathbf{z}_1^{(i)}, \dots, \mathbf{z}_{j-1}^{(i)}, \mathbf{z}_{j+1}^{(i-1)}, \dots, \mathbf{z}_J^{(i-1)}, \mathbf{z}_{J+1}, \dots, \mathbf{z}_P)$, meaning that at all times the most recently imputed values of all variables are used to impute other variables.

After repeating the sampling steps described by Equation (4.2) and (4.3) for every variable under imputation, the algorithm moves to the next iteration and repeats the same sampling steps for all variables under imputation. The convergence of the algorithm is usually assessed by plotting the trends of the average imputations across iterations for different starting values. After convergence, the imputations are assumed to be samples from the target multivariate distribution. With this process, one can generate as many imputed data sets as desired. Finally, the analysis model used to answer a substantive researcher question is estimated on each imputed data set, and the parameter estimates are pooled using Rubin's rules (Rubin, 1987).

For small values of P, the researcher imputing the data can use all of the columns in $\mathbf{Z}_{-j}$ as predictors in the univariate imputation model for $\mathbf{z}_j$. As P grows larger, the imputer needs to decide which predictors to include and which to leave out, a task that can require a considerable amount of expertise in both statistical modeling techniques and the field of the substantive research question. By summarizing the information in all of the possible predictors with a few linear combinations of the columns of $\mathbf{Z}_{-j}$, PCA and other dimensionality reduction techniques provide an accessible, data-driven way of specifying the imputation models.

4.2.2 Principal component analysis

PCA is a dimensionality reduction technique that finds a lower dimensional representation of the variables contained in an $N \times P$ data matrix $\mathbf{X}$ with minimal loss of information[19]. It does so by finding $Q \leq P$ independent linear combinations of the variables in the original data such that these linear combinations have maximum variance. One common formulation of PCA (e.g., Jolliffe, 2002; S. Park et al., 2021; Van Deun et al., 2011) models the data based on component weights as follows:

$$\mathbf{X} = \mathbf{X}\mathbf{W}\mathbf{P}' + \mathbf{E} \tag{4.4}$$

with $\mathbf{W}$ a $P \times Q$ matrix of weights, $\mathbf{P}$ a matrix of $P \times Q$ component loadings, and $\mathbf{E}$ an $N \times P$ matrix of residuals. The weights describe how the original data should be combined to form the Q linear combinations, or PCs ($\mathbf{T} = \mathbf{X}\mathbf{W}$, with $\mathbf{T}$ indicating an $N \times Q$ matrix of principal component scores). The loadings indicate how the original data can be reconstructed based on $\mathbf{T}$. The matrix of residual is the reconstruction error.

Based on the formulation in Equation (4.4), estimating PCA can be thought of as the task of finding the $\mathbf{W}$ and $\mathbf{P}$ minimizing the reconstruction error:

$$(\mathbf{W}, \mathbf{P}) = \underset{\mathbf{W},\mathbf{P}}{\text{argmin}} \; \|\mathbf{X} - \mathbf{X}\mathbf{W}\mathbf{P}'\|^2 \tag{4.5}$$

To make the solution identifiable, the constraint $\mathbf{W}'\mathbf{W} = \mathbf{I}$ is imposed. Usually, under this formulation, $\mathbf{W}$ is also restricted to be equal to $\mathbf{P}$, although this does not have to be the case (Zou et al., 2006). We wrote Equation (4.5) explicitly in terms of both $\mathbf{W}$ and $\mathbf{P}$ as this makes clear the connection between PCA and principal covariates regression, one of the SDR approaches we considered (see Section 4.2.4.2).

Because the component scores in $\mathbf{T}$ are summaries of the original data, they can be used as replacements for the original data in order to reduce the dimensionality of the data. For example, PCs can be used to replace a large number of predictors involved in a regression model to eliminate any possible collinearity. This approach is known as PCR. Specifically, given an outcome variable $\mathbf{y}$ and a set of P predictors $\mathbf{X}$, consider a standard regression model:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \epsilon, \tag{4.6}$$

where $\boldsymbol{\beta}$ is a $P \times 1$ vector of regression coefficients, and ϵ is a $N \times 1$ vector of independent normally distributed errors. With PCR we use Q PCs of $\mathbf{X}$ in its place so that Equation (4.6) can be rewritten as:

$$\mathbf{y} = \mathbf{T}\gamma + \epsilon, \tag{4.7}$$

---

[19] We follow the common practice of assuming that the columns of $\mathbf{X}$ are mean-centered and scaled to have a variance of 1.

where γ is a $Q \times 1$ vector of regression coefficients. The lower dimensionality of $\mathbf{T}$ compared to $\mathbf{X}$ and the independence of its columns allow Equation (4.7) to address the computational limitations of Equation (4.6) when P is large or when the variables in $\mathbf{X}$ are highly correlated.

The optimal number of components Q is never certain. A crucial feature of PCA is that the first PC explains the maximum amount of variance in all P predictors, the second PC explains the maximum variance of all P residuals, and so on. This means that the explained variance decreases as fast as the data allows as more PCs are retained. As a result, Q is usually taken to be much smaller than P; if $Q = P$, the variance in the P predictors would be redistributed across P new predictors and this would defy the goal of PCA to reduce the number P considerably while retaining as much variance as possible. In practice, when estimating PCR, researchers rely on cross-validation (e.g., Vervloet et al., 2016) to guide this decision.

4.2.3 Multiple imputation with principal component regression

Costantini et al. (2024) found that the best way to incorporate PCR into MICE is to extract PCs at every iteration. When imputing $\mathbf{z}_j$ in the ith iteration of MICE, the PCs can be estimated from $\mathbf{Z}_{-j}^{(i)}$ and used as predictors in the univariate imputation model. Each univariate imputation model can then be defined as:

$$f(\mathbf{z}_j | \mathbf{T}_{-j}^{(i)}, \boldsymbol{\theta}_j), \tag{4.8}$$

where $\mathbf{T}_{-j}^{(i)}$ is the matrix storing the PC scores estimated on $\mathbf{Z}_{-j}^{(i)}$. The steps described in Algorithm 1 are followed to impute $\mathbf{z}_j$ with PCR at every iteration. We refer to this use of PCR within MICE as MI-PCR. This MI-PCR incorporates uncertainty around the imputation model parameters using bootstrapping following the same principle as the 'imputation under the normal linear model with bootstrap' algorithm described by van Buuren (2018, p. 69).

MI-PCR allows the researcher imputating the data to include all predictors in the imputation model for every variable, bypassing the difficult model selection step, while preserving the advantages of an inclusive strategy. However, the performance of MI-PCR is highly sensitive to the number of PCs computed. Not using enough PCs to adequately represent the latent structure (i.e., using fewer PCs than the true number of latent variables) will produce poor imputations (Costantini et al., 2024). Furthermore, there is no guarantee the number of PCs that optimally represent $\mathbf{Z}_{-j}^{(i)}$ will be good imputation model predictors as the PCs retained might be summarizing information that is unrelated to the variable under imputation $\mathbf{z}_j$. Finally, performing

Algorithm 1 Imputation under the PCR model with bootstrap

```
For a given $\mathbf{z}_j$ variable under imputation, draw a bootstrap version $\mathbf{z}^*_{j,\text{obs}}$ with replacement from the observed cases $\mathbf{z}_{j,\text{obs}}$, and store as $\mathbf{Z}^*_{-j,\text{obs}}$ the corresponding rows of $\mathbf{Z}^{(i)}_{-j}$.
Center and scale $\mathbf{Z}^*_{-j,\text{obs}}$ and store the result as $\tilde{\mathbf{Z}}^*_{-j,\text{obs}}$
Center and scale $\mathbf{Z}^{(i)}_{-j,\text{mis}}$ based on the means and standard deviations of $\mathbf{Z}^*_{-j,\text{obs}}$ and store the result as $\tilde{\mathbf{Z}}_{-j,\text{mis}}$
Center $\mathbf{z}^*_{j,\text{obs}}$ on its mean value $\bar{\mathbf{z}}^*_{j,\text{obs}}$ and store it in $\tilde{\mathbf{z}}^*_{j,\text{obs}}$.
Estimate $\mathbf{W}$ and $\mathbf{P}$ by the eigendecomposition of the cross-product matrix of $\tilde{\mathbf{Z}}^*_{-j,\text{obs}}$
Compute the first $Q$ PCs as $\mathbf{T}^{(i)}_{-j,\text{obs}} = \tilde{\mathbf{Z}}^*_{-j,\text{obs}}\mathbf{W}$
Regress the mean-centered $\tilde{\mathbf{z}}^*_{j,\text{obs}}$ on $\mathbf{T}^{(i)}_{-j,\text{obs}}$ and store the regression coefficients $\boldsymbol{\beta}$
Estimate the residual error variance $\sigma^2$ as the ratio between the residual sum of square ($RSS$) and the degrees of freedom ($df$):
    $$\sigma^2 = RSS/df = \frac{\Sigma\,(\tilde{\mathbf{z}}^*_{j,\text{obs}} - \mathbf{T}^{(i)}_{-j,\text{obs}}\boldsymbol{\beta})^2}{N-Q}$$
Obtain the predicted values for $\mathbf{z}_{j,\text{mis}}$ by
    $$\hat{\mathbf{z}}_{j,\text{mis}} = \tilde{\mathbf{Z}}_{-j,\text{mis}}\mathbf{W}\boldsymbol{\beta}$$
Obtain imputations by adding normally distributed errors scaled by $\sigma^2$ to these predictions and by adding $\bar{\mathbf{z}}^*_{j,\text{obs}}$ to center them appropriately.
```

PCA on large data sets involves demanding matrix operations. MI-PCR requires repeating these intensive manipulations for every variable under imputation and every iteration of the MICE algorithm.

4.2.4 Multiple imputation with supervised dimensionality reduction

SDR techniques represent an alternative to PCA that could obviate some of the limitations of MI-PCR outlined above. In particular, the *supervision* of SDR methods should help computing PCs that are better predictors of the variables under imputation than the ones produced by PCA, and, as a corollary, it could also allow to retain fewer PCs than the number of latent variables in the data-generating model. In what follows we describe three alternative approaches to finding linear combinations of the predictors that do a good job of both summarizing the predictors and predicting the outcome variable. For each approach, we first describe how it works, and then we describe its implementation as a univariate imputation method in the MICE algorithm.

4.2.4.1 Supervised principal component regression

Bair et al. (2006) proposed computing the PCs only on the subset of variables that are associated with the dependent variable. Their approach is straightforward:

1. Regress $\mathbf{y}$ onto each column of $\mathbf{X}$ via P separate simple linear regressions. Because the data are standardized, the regression coefficients of these simple linear regression are equivalent to correlation coefficients. The strength of the association is what matters in the predictive task, so we consider only the absolute value of the correlation and refer to it as $\hat{\rho}$.
2. Define the subset $\mathbf{X}_s \in \mathbf{X}$ by discarding all variables whose correlation $\hat{\rho}$ is less than a selected threshold ρ_s.
3. Use $\mathbf{X}_s$ to estimate the PCs.
4. Use these PCs as independent variables in the PCR model.

A key aspect of the method is that both the number of PCs and the threshold value ρ_s can be determined by cross-validation. We refer to this approach as supervised principal component regression (SPCR).

In SPCR, the component weights are estimated by minimizing the same criterion as in Equation (4.5), but only a subset of relevant variables from $\mathbf{X}$ is used for the computation. By doing so, SPCR effectively sets to 0 component weights for variables that are not relevant predictors of $\mathbf{y}$. As a result, SPCR produces PCs that are better predictors of $\mathbf{y}$ and improves the predictive performance of PCR. We refer to the approach of excluding variables that are uninteresting for the prediction of the dependent variable as *discrete* supervision.

A similar approach, known as Sparse PCA (Zou et al., 2006), reduces the number of variables explicitly used in the PC computation by combining a lasso penalty with the PCA optimization criterion. Similarly to SPCR, Sparse PCA sets certain loadings to 0, but it does so to increase the interpretability of the resulting PCs, not to improve their predictive performance. This key difference makes SPCR a more suitable tool than Sparse PCA for aiding automatic imputation model specification. Therefore, we considered SPCR and not Sparse PCA as a means to reduce the dimensionality of the imputation models.

In the context of imputation, SPCR can be used as a univariate imputation model in a similar way to PCR. For each partially observed $\mathbf{z}_j$, with $j \in \{1, \dots, J\}$, the imputation model can be defined as:

$$f(\mathbf{z}_j | \mathbf{T}_s^{(i)}, \boldsymbol{\theta}_j), \tag{4.9}$$

where $\mathbf{T}_s^{(i)}$ is the matrix of PCs computed on $\mathbf{Z}_s^{(i)}$, the subset of variables with $\hat{\rho} > \rho_s$, at the ith iteration of the MICE algorithm. The steps described in Algorithm 2 are followed to impute $\mathbf{z}_j$ at every iteration. We refer to this use of SPCR within MICE as MI-SPCR.

In our implementation of MI-SPCR, *K*-fold cross-validation (Hastie et al., 2009, pp. 241–245) is used to select ρ_s, from a user-defined vector of possible values. For every threshold value in the interval $[0, 1]$, all predictors of $\mathbf{z}^*_{j,\text{obs}}$ in $\mathbf{Z}^*_{-j,\text{obs}}$ with a correlation larger than the threshold form an active set of predictors. Then, Q PCs are extracted from each active set and used to predict $\mathbf{z}^*_{j,\text{obs}}$ in a *K*-fold cross-validation procedure. The active set giving the lowest cross-validated prediction error is kept. As with MI-PCR, the number of components Q is considered fixed, but it can be selected by the same cross-validation procedure. Note that for a given number of components, only certain threshold values are allowed. We can compute Q components only if the data have at least Q columns. Therefore, the more components we want to estimate, the less restrictive ρ_s can be. If we ask for as many components as there are columns in the data, then ρ_s must be large enough to keep all columns of the data, making MI-SPCR equivalent to MI-PCR.

4.2.4.2 Principal covariates regression

Principal covariates regression (PCovR, de Jong & Kiers, 1992) is an SDR approach that modifies the optimization criteria behind PCA to include information from the outcome variable in the optimization problem. PCovR looks for a lower dimensional representation of $\mathbf{X}$ that accounts for the maximum amount of variation in both $\mathbf{X}$ and $\mathbf{y}$. To understand how PCovR differs from PCR consider the following decomposition of the data:

$$\mathbf{X} = \mathbf{T}\mathbf{P}'_{\mathbf{X}} + \mathbf{E}_{\mathbf{X}} \tag{4.10}$$

$$\mathbf{y} = \mathbf{T}\mathbf{P}'_{\mathbf{y}} + \mathbf{e}_{\mathbf{y}} \tag{4.11}$$

$$\mathbf{T} = \mathbf{X}\mathbf{W} \tag{4.12}$$

Algorithm 2 Imputation under the SPCR model with bootstrap

```
For a given $\mathbf{z}_j$ variable under imputation, draw a bootstrap version $\mathbf{z}^*_{j,\text{obs}}$ with replacement from the observed cases $\mathbf{z}_{j,\text{obs}}$, and store as $\mathbf{Z}^*_{-j,\text{obs}}$ the corresponding values on $\mathbf{Z}^{(i)}_{-j,\text{obs}}$.
Compute the absolute correlation between $\mathbf{z}^*_{j,\text{obs}}$ and every potential predictor in $\mathbf{Z}^*_{-j,\text{obs}}$.
For every $\rho_h$, create a set of predictors with absolute correlation higher than $\rho_h$.
Use $K$-fold cross-validation to select the value of $\rho_h$ and the associated set of predictors that return the PCR model with the smallest prediction error. Define $\rho_s$ as the selected value.
Drop from $\mathbf{Z}^*_{-j,\text{obs}}$ and $\mathbf{Z}^{(i)}_{-j,\text{mis}}$ all variables with an absolute correlation smaller than $\rho_s$ and create $\mathbf{Z}^*_{s,\text{obs}}$ and $\mathbf{Z}_{s,\text{mis}}$.
Center and scale $\mathbf{Z}^*_{s,\text{obs}}$ and store the result as $\tilde{\mathbf{Z}}^*_{s,\text{obs}}$
Center and scale $\mathbf{Z}_{s,\text{mis}}$ based on based on the means and standard deviations of $\mathbf{Z}^*_{s,\text{obs}}$ and store the result as $\tilde{\mathbf{Z}}_{s,\text{mis}}$
Center $\mathbf{z}^*_{j,\text{obs}}$ on its mean value $\bar{\mathbf{z}}^*_{j,\text{obs}}$ and store it in $\tilde{\mathbf{z}}^*_{j,\text{obs}}$.
Estimate $\mathbf{W}$ and $\mathbf{P}$ by the eigendecomposition of the cross-product matrix of $\tilde{\mathbf{Z}}^*_{s,\text{obs}}$
Compute the first $Q$ PCs as $\mathbf{T}^{(i)}_{s,\text{obs}} = \tilde{\mathbf{Z}}^*_{s,\text{obs}}\mathbf{W}$
Regress the mean centered $\tilde{\mathbf{z}}^*_{j,\text{obs}}$ on $\mathbf{T}^{(i)}_{s,\text{obs}}$ and store the regression coefficients $\boldsymbol{\beta}$.
Estimate the residual error variance $\sigma^2$ as the ratio between the residual sum of square ($RSS$) and the degrees of freedom ($df$):
    $\sigma^2 = RSS/df = \frac{\Sigma\,(\tilde{\mathbf{z}}^*_{j,\text{obs}} - \mathbf{T}^{(i)}_{s,\text{obs}}\boldsymbol{\beta})^2}{N-Q}$
Obtain the predicted values for $\mathbf{z}_{j,\text{mis}}$ by
    $\hat{\mathbf{z}}_{j,\text{mis}} = \tilde{\mathbf{Z}}_{s,\text{mis}}\mathbf{W}\boldsymbol{\beta}$
Obtain imputations by adding normally distributed errors scaled by $\sigma^2$ to these predictions and by adding $\bar{\mathbf{z}}^*_{j,\text{obs}}$ to center them appropriately.
```

where $\mathbf{T}$ and $\mathbf{W}$ are defined as in (4.5), $\mathbf{P_X}$ is $\mathbf{P}$ from (4.5), and $\mathbf{P_y}$ is the $Q \times 1$ vector of weights relating $\mathbf{y}$ to the component scores in $\mathbf{T}$. $\mathbf{E_X}$ and $\mathbf{e_y}$ are reconstruction errors. They represent the information lost by using $\mathbf{T}$ as a summary of $\mathbf{X}$ and the errors in the linear regression model, respectively. PCovR can be formulated as the task of minimizing a weighted combination of both $\mathbf{E_X}$ and $\mathbf{e_y}$:

$$(\mathbf{W}, \mathbf{P_X}, \mathbf{P_y}) = \underset{\mathbf{W}, \mathbf{P_X}, \mathbf{P_y}}{\text{argmin}} \ \alpha \left\|(\mathbf{X} - \mathbf{XWP'_X})\right\|^2 + (1 - \alpha) \left\|(y - \mathbf{XWP'_y})\right\|^2 \qquad (4.13)$$

subject to the constraint $(\mathbf{WX})'\mathbf{XW} = \mathbf{T}'\mathbf{T} = \mathbf{I}$.

The α parameter defines which reconstruction error is being prioritized. When $\alpha = 1$, the emphasis is exclusively placed on reconstructing $\mathbf{X}$, casting PCR as a special case of PCovR. When $\alpha = 0.5$, the importance of $\mathbf{X}$ and $\mathbf{y}$ is equally weighted, a case that resembles partial least square regression (PLSR), which we discuss in subsection 4.2.4.3. In practice, the value of α can be found by cross-validation or according to a sequential procedure based on maximum likelihood principles (Vervloet et al., 2013). In particular,

$$\alpha_{ML} = \frac{\|\mathbf{X}\|^2}{\|\mathbf{X}\|^2 + \|\mathbf{y}\|^2 \frac{\hat{\sigma}^2_{\mathbf{E_X}}}{\hat{\sigma}^2_{e_\mathbf{y}}}} \qquad (4.14)$$

where $\hat{\sigma}^2_{\mathbf{E_X}}$ can be obtained as the unexplained variance by components computed according to classical PCA on $\mathbf{X}$ and $\hat{\sigma}^2_{e_\mathbf{y}}$ can be estimated as the unexplained variance by the linear model regressing $\mathbf{y}$ on $\mathbf{X}$. Note that, for the same data set, the more components are retained, the smaller $\hat{\sigma}_{\mathbf{E_X}}$ is, the higher α_{ML} is, and the closer PCovR becomes to PCR. Retaining the same number of components as the number of variables in $\mathbf{X}$ results in $\hat{\sigma}_{\mathbf{E_X}} = 0$ and $\alpha = 1$, which casts PCR as a special case of PCovR.

Compared to PCR, PCovR allows estimating PCs that not only represent well the predictor variables but also predict well the dependent variable. Compared to SPCR, the PCs computed with PCovR are always linear combinations of *all* variables in $\mathbf{X}$. PCovR can downweigh irrelevant variables for the prediction of $\mathbf{y}$, but it will never exclude them entirely. We refer to the PCovR approach to supervision as *continuous*, as opposed to the *discrete* supervision of SPCR.

When applied to the MICE algorithm, we can use PCovR as a univariate imputation model in a similar way to how we can use PCR and SPCR. At every iteration of the MICE algorithm, the steps described in Algorithm 3 are followed to impute $\mathbf{z}_j$. We refer to this use of PCovR within MICE as MI-PCovR.

Algorithm 3 Imputation under the PCovR model with bootstrap

```
For a given $\mathbf{z}_j$ variable under imputation, draw a bootstrap version $\mathbf{z}^*_{j,\text{obs}}$ with replacement from the observed cases $\mathbf{z}_{j,\text{obs}}$, and store as $\mathbf{Z}^*_{-j,\text{obs}}$ the corresponding values on $\mathbf{Z}^{(i)}_{-j}$.
Center and scale $\mathbf{Z}^*_{-j,\text{obs}}$ and store the result as $\tilde{\mathbf{Z}}^*_{-j,\text{obs}}$.
Center and scale $\mathbf{Z}^{(i)}_{-j,\text{mis}}$ based on based on the means and standard deviations of $\mathbf{Z}^*_{-j,\text{obs}}$ and store the result as $\tilde{\mathbf{Z}}_{-j,\text{mis}}$.
Center $\mathbf{z}^*_{j,\text{obs}}$ on its mean value $\bar{\mathbf{z}}^*_{j,\text{obs}}$ and store it in $\tilde{\mathbf{z}}^*_{j,\text{obs}}$.
Compute the value of $\alpha$ based on (4.14).
Compute $Q$ PCs by estimating the PCovR $\mathbf{W}$, $\mathbf{P}_{-j}$, and $\mathbf{P}_j$ based on $\tilde{\mathbf{z}}^*_{j,\text{obs}}$ and $\tilde{\mathbf{Z}}^*_{-j,\text{obs}}$. Note that $\mathbf{P}_{-j}$, and $\mathbf{P}_j$ correspond to $\mathbf{P}_{\mathbf{X}}$, and $\mathbf{P}_{\mathbf{y}}$ of Equation (4.13).
Estimate the residual error variance $\sigma^2$ as the ratio between the residual sum of square ($RSS$) and the degrees of freedom ($df$):
    $$\begin{aligned} \sigma^2 &= RSS/df \\ &= \frac{\Sigma\,(\tilde{\mathbf{z}}^*_{j,\text{obs}} - \tilde{\mathbf{Z}}^*_{-j,\text{obs}}\mathbf{W}\mathbf{P}'_j)^2}{N-Q} \end{aligned}$$
Obtain the predicted values for $\mathbf{z}_{j,\text{mis}}$ by
    $$\hat{\mathbf{z}}_{j,\text{mis}} = \tilde{\mathbf{Z}}_{-j,\text{mis}}\mathbf{W}\mathbf{P}'_{-j}$$
Obtain imputations by adding normally distributed errors scaled by $\sigma^2$ to these predictions and by adding $\bar{\mathbf{z}}^*_{j,\text{obs}}$ to center them appropriately.
```

4.2.4.3 Partial least square regression

Partial least square regression (PLS, Wold, 1975) is a dimensionality reduction technique that seeks linear combinations (or PLS components) that account for a large proportion of the variance in the predictors and correlate strongly with the dependent variable. Like PCR, PLSR finds independent linear combinations of the predictors in $\mathbf{X}$ that summarize the data well and uses these linear combinations to predict the dependent variable. Like PCovR, the weights defining the linear combinations of the predictors are computed using all the predictor variables and the outcome, resulting in a *continuous* supervision: irrelevant variables will be downweighted in the linear combinations, but they will not be completely ignored. Unlike PCR, SPCR, and PCovR, PLSR computes one linear combination at a time and stops at the required number of PLS components Q.

PLSR estimates $\mathbf{t}_1$, the first PLS component, by:

1. Computing the vector of weights $\mathbf{w}_1$ with elements $w_{1,j} = \mathbf{x}_j'\mathbf{y}$ for $j \in \{1, \dots, P\}$, the inner products between each predictor $\mathbf{x}_j$ and the dependent variable $\mathbf{y}$.
2. Deriving the constructed variable $\mathbf{t}_1 = \sum_{j=1}^{P} \mathbf{x}_j w_{1,j} = \mathbf{X}\mathbf{w}_1$.
3. Orthogonalizing $\mathbf{x}_1$ to $\mathbf{x}_P$ with respect to $\mathbf{t}_1$.

The second linear combination ($\mathbf{t}_2 = \mathbf{X}\mathbf{w}_2$) is then derived by repeating the same procedure but replacing each $\mathbf{x}_j$ with their versions orthogonalized with respect to $\mathbf{t}_1$. In PLSR, the qth weight vector ($\mathbf{w}_q$) maximizes the following optimization criterion (Frank & Friedman, 1993; Stone & Brooks, 1990):

$$\underset{\mathbf{w}_q}{\text{argmax}}\ \text{Corr}^2(\mathbf{y}, \mathbf{X}\mathbf{w}_q)\text{Var}(\mathbf{X}\mathbf{w}_q) \tag{4.15}$$

where $\text{Corr}^2(.)$ is the squared correlation of the vectors between brackets. As with all other methods, the linear combinations derived by the PLS algorithm are constrained to be mutually orthogonal.

As with SPCR and PCovR, at every iteration of the MICE algorithm, we can use PLSR to obtain imputations. Algorithm 4 describes the univariate imputation method based on PLSR[20] that we used to impute $\mathbf{z}_j$. We refer to this use of PLSR within MICE as MI-PLSR. In Table 4.1, we summarize the differences between the univariate imputation methods used by MI-PCR and the SDR-based approaches we described.

Algorithm 4 Imputation under the PLSR model with bootstrap

```
1: For a given $\mathbf{z}_j$ variable under imputation, draw a bootstrap version $\mathbf{z}^*_{j,\text{obs}}$ with replacement from the observed cases $\mathbf{z}_{j,\text{obs}}$, and store as $\mathbf{Z}^*_{-j,\text{obs}}$ the corresponding values on $\mathbf{Z}^{(i)}_{-j}$.
2: Estimate PLSR with $Q$ components by regressing $\mathbf{z}^*_{j,\text{obs}}$ onto $\mathbf{Z}^*_{-j,\text{obs}}$.
3: Estimate the residual error variance $\sigma^2$ as the ratio between the residual sum of square ($RSS$) and the degrees of freedom ($df$).
4: Obtain the predicted values for $\mathbf{z}_{j,\text{mis}}$ based on the trained PLSR model.
5: Obtain imputations by adding noise scaled by $\sigma^2$ to these predictions.
```

---

[20] A similar version of PLS as a univariate imputation method has also been implemented in the R package *miceadds* (Robitzsch & Grund, 2022).

Method	Supervision type	Optimization criterion	Estimated parameters	Tuning parameters
MI-PCR	none	$\underset{\mathbf{W},\mathbf{P}}{\text{argmin}}\ \lVert \tilde{\mathbf{Z}}^*_{-j,\text{obs}} - \tilde{\mathbf{Z}}^*_{-j,\text{obs}}\mathbf{W}\mathbf{P}' \rVert^2$	$\mathbf{W}, \mathbf{P}$	-
MI-SPCR	discrete	$\underset{\mathbf{W},\mathbf{P}}{\text{argmin}}\ \lVert \tilde{\mathbf{Z}}^*_{s,\text{obs}} - \tilde{\mathbf{Z}}^*_{s,\text{obs}}\mathbf{W}\mathbf{P}' \rVert^2$	$\mathbf{W}, \mathbf{P}$	ρ_s
MI-PCovR	continuous	$\underset{\mathbf{W},\mathbf{P}_\mathbf{X},\mathbf{P}_\mathbf{y}}{\text{argmin}}\ \alpha \lVert (\tilde{\mathbf{Z}}^*_{-j,\text{obs}} - \tilde{\mathbf{Z}}^*_{-j,\text{obs}}\mathbf{W}\mathbf{P}'_\mathbf{X}) \rVert^2 + (1-\alpha) \lVert (\tilde{\mathbf{z}}^*_{j,\text{obs}} - \tilde{\mathbf{Z}}^*_{-j,\text{obs}}\mathbf{W}\mathbf{P}'_\mathbf{y}) \rVert^2$	$\mathbf{W}, \mathbf{P}_\mathbf{X}, \mathbf{P}_\mathbf{y}$	α
MI-PLSR	continuous	$\underset{\mathbf{w}_q}{\text{argmax}}\ \text{Corr}^2(\tilde{\mathbf{z}}^*_{j,\text{obs}}, \tilde{\mathbf{Z}}^*_{-j,\text{obs}}\mathbf{w}_q)\text{Var}(\tilde{\mathbf{Z}}^*_{-j,\text{obs}}\mathbf{w}_q)$	$\mathbf{w}_q$	-

Table 4.1: Summary of the differences between the univariate imputation models used within the MICE algorithm.

Experimental factor	Label	Levels
Number of latent variables	L	2, 10, 50
Missing data mechanism used	$mech$	MCAR, MAR
Proportion of missing values	pm	0.1, 0.25, 0 50
Missing data treatment	$method$	MI-PCR, MI-SPCR, MI-PCovR, MI-PLS, MI-QP, MI-AM, MI-ALL, CC, FO
Number of components	nc	0, 1 to 12, 20, 29, 30, 40, 48, 49, 50, 51, 52, 60, 149

Table 4.2: Summary of experimental factors for the simulation study.

4.3 Simulation study

We investigated the relative performance of unsupervised PCR and the supervised alternatives described above with a Monte Carlo simulation study. In particular, we investigated the estimation bias, confidence interval width, and confidence interval coverage of several analysis model parameters obtained after imputation. We varied the dimensionality of the latent structure in the data-generating model, the proportion of missing values, the missing data mechanism, and the number of components used as predictors in the imputation models. The proportion of missing values and the missing data mechanism both influence the statistical properties of any missing data treatment, while the number of latent variables in the data-generating model and the number of components used in the imputation model affect the extent to which supervision can improve upon the limitations of MI-PCR. In Table 4.2, we summarize the experimental factors we varied and the levels we used with each factor.

4.3.1 Procedure

The simulation study involved four steps:

1. Data generation: We generated $R = 240$ data sets from a confirmatory factor analysis model, following the procedure described in Section 4.3.1.1.
2. Missing data generation: We generated missing values on three target items in each data set, following the procedure described in Section 4.3.1.2.
3. Imputation: We generated $d = 5$ multiply imputed versions of each generated data set using different imputation methods, as described in Section 4.3.1.3.

4. Analysis: We estimated the means, variances, covariances, and correlations of the three items with missing values on the d imputed data sets, and we pooled the estimates according to Rubin's rules (Rubin, 1986, p. 76). We then assessed each imputation method by computing the bias of different parameter estimates, and their confidence interval widths and coverages as described in Section 4.3.1.4.

4.3.1.1 Data generation

For each of the R replications, we generated a $1000 \times P$ data matrix $\mathbf{Z}$. The sample size should be large enough to generate data sets that have statistical properties similar to large social science data sets. Each data set was generated based on the following model:

$$\mathbf{Z} = \mathbf{F}\mathbf{\Lambda}' + \mathbf{E}, \tag{4.16}$$

where $\mathbf{F}$ is a $1000 \times L$ matrix of latent variables scores, L is the number of latent variables, $\mathbf{\Lambda}$ is a $3 \times L$ matrix of factor loadings, where 3 is the number of items measuring each latent variable, and $\mathbf{E}$ is a $1000 \times P$ matrix of measurement errors, where $P = 3 * L$. The dimensionality of the data resembles that of the many large social surveys that use short scales to measure respondents' attitudes such as *political engagement* and *anti-immigrant attitudes*. For example, consider the European Values Study (EVS, 2020a) which measures a variety of attitudes with 3, 4, or 5 items.

The factor loading matrix $\mathbf{\Lambda}$ in Equation (4.16) described a simple structure (Bollen, 1989, p. 234) where each item loads on exactly one of the L latent variables. In real data applications, this factor structure is uncommon but not implausible. For example, a relatively clear simple structure can be found when analyzing personality inventories (e.g., NEO-PR-I, Costa Jr et al., 1991) with the *Neuroticism-Extraversion-Openness* three-factor model (McCrae & Costa Jr, 1983). A simple structure is also often assumed when performing exploratory factor analysis because it provides the most parsimonious explanation (Costa Jr & McCrae, 2008, pp. 183-184).

We sampled $\mathbf{F}$ from a multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{\Psi}$. The correlation between the first and second latent variables was fixed at 0.8, while the correlation between all other latent variables and the first two was fixed at 0.1. Together with factor loadings fixed at $\lambda = 0.85$, these choices resulted in correlations of approximately 0.72, 0.58, and 0.07 between items measuring the same latent variable, items measuring the first and second latent variable, and items measuring the first latent variable and the others, respectively. These values represent plausible, but reasonably high, item-scale associations and they should mitigate the impact of measurement error on our findings without resorting to implausibly precise data.

The matrix of measurement errors $\mathbf{E}$ was sampled from a multivariate normal distribution with mean vector $\mathbf{0}$ and covariance matrix $\mathbf{\Theta}$. The off-diagonal elements of $\mathbf{\Theta}$ were set to 0 to reflect uncorrelated errors, while the diagonal elements were specified as $1 - \lambda^2$ to give the simulated items unit variances. After sampling, the columns of $\mathbf{Z}$ were rescaled to have approximately a mean of 5 and a variance of 6.5, which are common values for Likert items in social surveys measured on a 10-point scale (for example in EVS, 2020a).

In this data-generating procedure, we considered the number of latent variables used (L) as an experimental factor with levels $2, 10, 50$, resulting in data sets containing 6, 30, and 150 total items. Costantini et al. (2024) showed that the number of components used in MI-PCR needs to be at least as high as the number of latent variables in the data-generating model. So, we expected MI-PCR to require at least L components to achieve satisfactory performance. One of this study's main objectives was to understand how well supervision can overcome this limitation of MI-PCR. We generated data according to a confirmatory factor analysis model instead of generating the data directly based on true principal components to avoid the misleading results that can occur when using a single model for both data generation and imputation (Oberman & Vink, 2023, p. 4).

4.3.1.2 Missing data generation

We generated missing data on the three items measuring the first latent variable ($\mathbf{z}_1$, $\mathbf{z}_2$, $\mathbf{z}_3$). The proportion of missing values per variable (pm) was defined as an experimental factor taking three levels $pm \in \{0.1, 0.25, 0.5\}$. The missing data mechanism ($mech$) was a factor with two levels:

- Missing completely at random (MCAR): To test how the methods performed in the simplest possible missing data mechanism, we generated missing values on each item based on a missing data indicator (δ) sampled from a binomial distribution with success probability pm. If $\delta = 1$, the item score was set to missing. If $\delta = 0$, the item score was set to observed. As a result, every variable had a proportion of missing values approximately equal to pm.

- Missing at random (MAR): To test how the methods performed in a more realistic situation, we generated missing values based on a MAR mechanism with the three items measuring the second latent variable ($\mathbf{z}_4$, $\mathbf{z}_5$, $\mathbf{z}_6$) used as predictors of missingness. We sampled δ from Bernoulli distributions with probabilities defined based on the following logit model:

$$logit(\delta = 1) = \beta_0 + \mathbf{Z}_{(4,5,6)}\boldsymbol{\beta}, \tag{4.17}$$

where β_0 is an intercept parameter, and $\boldsymbol{\beta}$ is a vector of slope parameters. All slopes in $\boldsymbol{\beta}$ were fixed to 1, while the value of β_0 was chosen with an optimization algorithm that minimized the difference between the actual and desired proportion of missing values on the variable. The pseudo R-squared for the logistic regression of the missing value indicator on the predictors of missingness was approximately 14%. The AUC for the logistic regression was approximately 0.74. To create realistic missing data patterns, the location of missing data was fixed to right for $\mathbf{z}_1$, left for $\mathbf{z}_2$, and tails for $\mathbf{z}_3$.

4.3.1.3 Imputation

We imputed the missing values using the four dimension reduction-based methods described above (MI-PCR, MI-SPCR, MI-PCovR, MI-PLSR) as well as three traditional approaches:

- MI with all the available variables used as predictors in the imputation models (MI-ALL), which represents the most naive way to define the imputation model.
- MI with a correlation-based threshold strategy to select the subset of important predictors (MI-QP). As a pragmatic point of comparison, this method used the *quickpred* function from the R package *mice* (van Buuren & Groothuis-Oudshoorn, 2011) to select the predictors for the univariate imputation models via the correlation-based threshold strategy described by van Buuren et al. (1999, pp. 687–688). To implement this approach, we selected only those predictors that correlated with the imputation targets (or their associated missingness indicators) higher than 0.1.
- MI with the analysis model variables used as sole predictors in the imputation models (MI-AM). This method produces the simplest possible congenial imputation model and is interesting due to its popularity in the social scientific literature (Costantini, Lang, Reeskens, & Sijtsma, 2023).

As reference points, we also treated the missing values with complete-case analysis (CC) and estimated the analysis model from the original, fully observed data.

Every imputation-based method used simple random draws from the observed data as starting values and was run to obtain 5 imputed data sets. Convergence was achieved after 20 iterations for all methods[21]. All our dimensionality reduction-based algorithms need the user to define Q, the optimal number of components. In practice, researchers will choose a single value of Q with a cross-validation procedure, but in this study, we are interested in evaluating the performance of the four dimension-reduction imputation approaches while varying the number of retained components. Therefore, we defined Q as one of our main experimental factors (the number of components, nc) taking values $\{1, \dots, 12, 20, 29, 30, 40, 48, 49, 50, 51, 52, 60, 149\}$. These values were chosen to both cover the range of possible choices (i.e., $1, \dots, [P-1]$) and to provide more granularity around the true number of latent variables (i.e., 3, 10, 50).

Finally, for MI-SPCR, we used the cross-validation procedure described in Section 4.2.4.1 to select ρ_s from the vector of values $\{0.05, 0.1, 0.15, \dots, 0.95\}$. For a given number of components, some threshold values can exclude enough predictors to preclude computing the required number of components. To avoid this possibility, the cross-validation algorithm only considered values of ρ that retained enough variables to compute the required number of components. The weighting parameter (α) for MI-PCovR was selected using the sequential MLE-based estimation procedure described in Vervloet et al. (2013). The degrees of freedom for the PLSR imputation model were computed based on the naive approach described by Krämer and Sugiyama (2011).

4.3.1.4 Analysis and comparison criteria

For a given parameter ϕ (e.g., the mean of $\mathbf{z}_1$, the correlation between $\mathbf{z}_1$ and $\mathbf{z}_2$), we used the absolute percent relative bias (PRB) to quantify the estimation bias introduced by the imputation procedure:

$$\text{PRB} = \left| \frac{\bar{\hat{\phi}} - \phi}{\phi} \right| \times 100 \tag{4.18}$$

where ϕ is the true value of the focal parameter defined as $\sum_{r=1}^{R} \hat{\phi}_r^{full}/R$, with ϕ_r^{full} being the parameter estimate for the rth repetition computed on the fully observed data. The averaged focal parameter estimate under a given missing data treatment was computed as $\bar{\hat{\phi}} = \sum_{r=1}^{R} \hat{\phi}_r/R$, with $\bar{\hat{\phi}}_r$ being the estimate obtained from the

---

[21] Convergence plots are reported in the interactive results dashboard that we developed to accompany this article. See Section 4.3.2 for more details.

treated incomplete data in the rth replication. Following Muthén et al. (1987), we considered PRB > 10 as indicative of problematic estimation bias.

To measure the statistical efficiency of the imputation methods we computed the average width of the confidence intervals (CIW).

$$\text{CIW} = \frac{\sum_{r=1}^{R}(\widehat{\text{CI}}_r^{upper} - \widehat{\text{CI}}_r^{lower})}{R}, \tag{4.19}$$

with $\widehat{\text{CI}}_r^{upper}$ and $\widehat{\text{CI}}_r^{lower}$ being the upper and lower bounds of the estimated confidence interval for the rth replication. Narrower CIWs indicate higher efficiency. However, narrower CIWs are not preferred if they come at the expense of good confidence interval coverage (CIC) of the parameter values. CIC is the proportion of confidence intervals that contain the true value of the parameter, across the R data samples:

$$\text{CIC} = \frac{\sum_{r=1}^{R} I(\phi \in \widehat{\text{CI}}_r)}{R}, \tag{4.20}$$

where $\widehat{\text{CI}}_r$ is the confidence interval of the parameter estimate $\hat{\phi}_r$ in the rth replication, and $I(.)$ is the indicator function that returns 1 if the argument is true and 0 otherwise. CIC depends on both the bias and the variability of the CIW for a parameter estimate. In particular, for a given level of bias, a narrower CIW leads to lower CIC, and, for a given CIW, a larger bias leads to lower CIC. An imputation method with good coverage should result in CICs greater than or equal to the nominal rate. For 95% CIs, CIC below 0.9 is usually considered problematic (e.g., van Buuren, 2018, p. 52; Collins et al., 2001, p. 340) as it implies inflated Type I error rates. High CIC (e.g., 0.99) implies inflated Type II error rates.

4.3.2 Results

We report only the results for the correlation between $\mathbf{z}_1$ and $\mathbf{z}_2$ for the conditions with *mech* = MAR and $pm = 0.5$ because the type of parameter and the different levels of these two factors did not impact the relative performances of the imputation methods. We focused on the correlation between two items with missing values because this parameter differentiated the performances of the methods the most. The full set of results is available via the interactive results dashboard that we developed to accompany this article (Costantini, 2023g). The dashboard can be downloaded and installed as an R package, and it can be used as an R Shiny app.

In Figures 4.1, 4.2, and 4.3, we report the PRB, CIW, and CIC for the correlation coefficient between $\mathbf{z}_1$ and $\mathbf{z}_2$ for different numbers of latent variables in the data-generating model (L) and numbers of components retained by the methods (nc).

Across all values of L, MI-PCR resulted in a smaller bias and coverage closer to nominal the more components were retained. However, MI-PCR required the number of components to be greater or equal to the number of latent variables used in the data-generating model to return acceptable bias and coverages. In particular, for $L = 2$ and $L = 10$, MI-PCR resulted in acceptable bias (PRB < 10) and close to nominal coverage (CIC > 0.9) only when using $nc \geq 2$ and $nc \geq 10$, respectively. Contrary to expectation, this trend did not persist for all higher values of L and nc. For $L = 50$, MI-PCR resulted in high bias (PRB > 20) even for $nc = 50$. Furthermore, for $L = 2$ and $L = 10$, MI-PCR resulted in large deviations from nominal coverage (CIC < 0.9) for $nc = 5$ and for $nc \in \{11, 12\}$, respectively.

Compared to MI-PCR, MI-SPCR performed much better, especially when using just a few components. MI-SPCR resulted in the lowest bias, smallest confidence interval width, and closest to nominal coverage when using between 2 and 5 components. For all values of L, using 2 components instead of 1, led to a large reduction in PRB and improvement in CIC. Using 6 or more components had only a minor negative impact on the performance of the method in the condition with $L = 10$, resulting in a negligible increase in bias. However, for $L = 50$, using 6 or more components did lead to high bias and low coverage. Finally, the maximum number of components led to algorithmic failures, so there were no results to report for $nc = 29$ and $nc = 149$ in the $L = 10$ and $L = 50$ conditions, respectively.

MI-PCovR resulted in acceptable bias for all values of L, for all nc values reported. However, its bias performance was less stable than that of MI-SPCR across the values of L and nc. For $L = 10$, MI-PCovR led to smaller bias when using $nc = 2$ instead of $nc = 1$, but the bias increased when using $nc \in \{3, \dots, 9\}$, only to decrease again for $nc \geq 10$. In the $L = 50$ condition, MI-PCovR resulted in decreasing bias for the range $nc \in \{1, \dots, 9\}$, and the lowest bias was achieved with $nc = L$, just after a small increase. Furthermore, the CIC resulted in larger deviations from nominal coverage than MI-SPCR, resulting in acceptable CIC only with a few of the many nc values considered.

Compared to MI-SPCR and MI-PCovR, the reduction in bias obtained by MI-PLSR for higher values of nc was more gradual. For $L = 10$ and 50, using 2 components instead of 1, led to a reduction in bias, but 3 components were necessary to achieve PRB < 10, while both MI-SPCR, and MI-PCovR only needed 2 to achieve the same result. However, the PRB remained small for $nc \in \{6, 7, 8, 9, 10\}$, even when that of MI-SPCR and MI-PCovR increased. Despite this good bias performance, the CIW and CIC of MI-PLSR fluctuated between acceptable and not, with only a few values of nc resulting in close-to-nominal coverage.

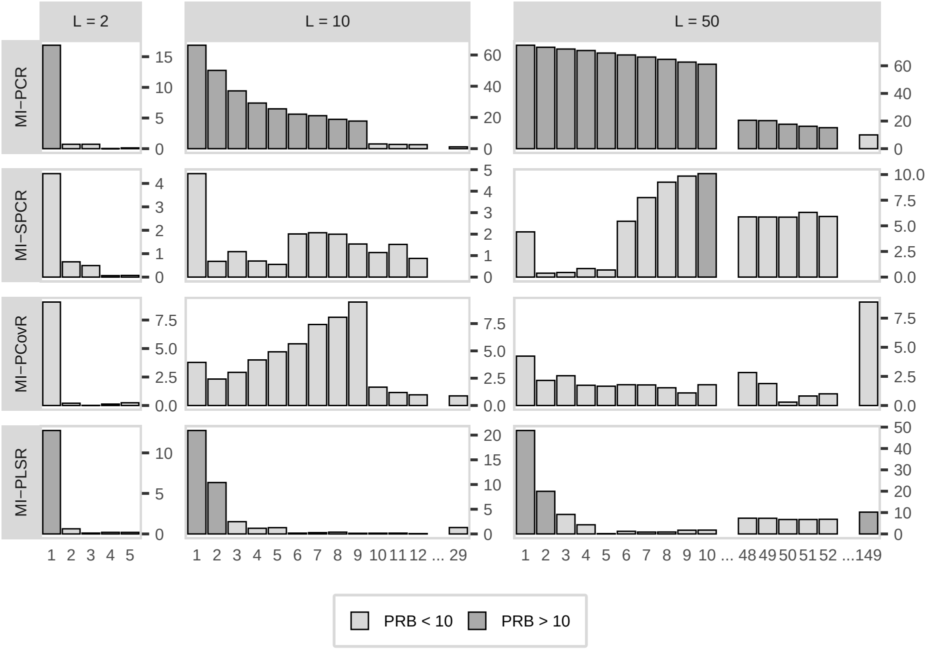


Figure 4.1: The PRB for the estimated correlation coefficient between the first two items imputed is reported (Y-axis) as a function of the number of components (nc) used by the PCA-based imputation methods (X-axis). The plot is divided into a grid where the rows distinguish the results obtained after imputing the data with the four PCA-based methods and the columns distinguish the number of latent variables used to generate the data (L). All results plotted in this figure were obtained on data generated with *mech* = MAR and $pm = 0.5$.

To put these results into perspective, we reported the same performance metrics for three traditional MI approaches, and complete-case analysis in Figure 4.4. MI-QP and MI-ALL resulted in acceptable bias ($PRB < 10$) for all values of L, although larger values of L did result in increased bias and decreasing coverage for both methods, but especially for MI-ALL. MI-AM and CC did not result in a higher bias or lower coverages for larger values of L, but both returned relatively high bias and low coverage across all conditions.

4.4 Discussion

4.4.1 Supervised dimensionality reduction

Our simulation study outlined some clear advantages of using supervised dimensionality reduction techniques over standard PCA with MICE. We found that MI-PCR

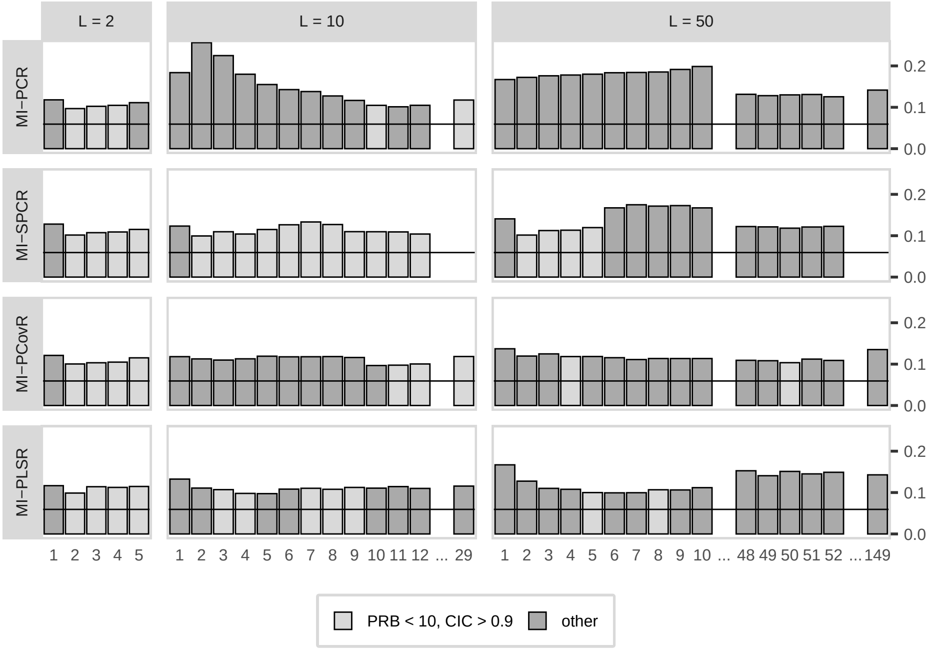


Figure 4.2: The CIW for the estimated correlation coefficient between the first two items imputed is reported (Y-axis) as a function of the number of components (nc) used by the PCA-based imputation methods (X-axis). The plot is divided into a grid where the rows distinguish the results obtained after imputing the data with the four PCA-based methods and the columns distinguish the number of latent variables used to generate the data (L). All results plotted in this figure were obtained on data generated with *mech* = MAR and $pm = 0.5$. The light gray color indicates the parameter estimate for which the CIW is reported had both acceptable bias (PRB < 10) and coverage (CIC > 0.9). The dark gray color indicates the parameter estimate for which the CIW is reported had large bias (PRB > 10) or low coverage (CIC < 0.9), or both. The black horizontal lines represent the average CIW obtained on the original fully observed data.

requires the use of at least as many components as the number of latent variables in the data-generating model, which is in line with the results presented by Costantini et al. (2024). The simulation study presented here also showed that meeting this requirement is not sufficient to obtain good imputations for a large number of latent variables. We found that the performance of MI-PCR does not only depend on knowing the number of latent variables in the data-generating model, but also on the number itself. On the contrary, the SDR-based methods retaining just a few components resulted in small bias and good confidence interval coverage, *independent* of the number of latent variables in the data-generating model. Furthermore, when using any given number of components, except the maximum, using the SDR-based

]

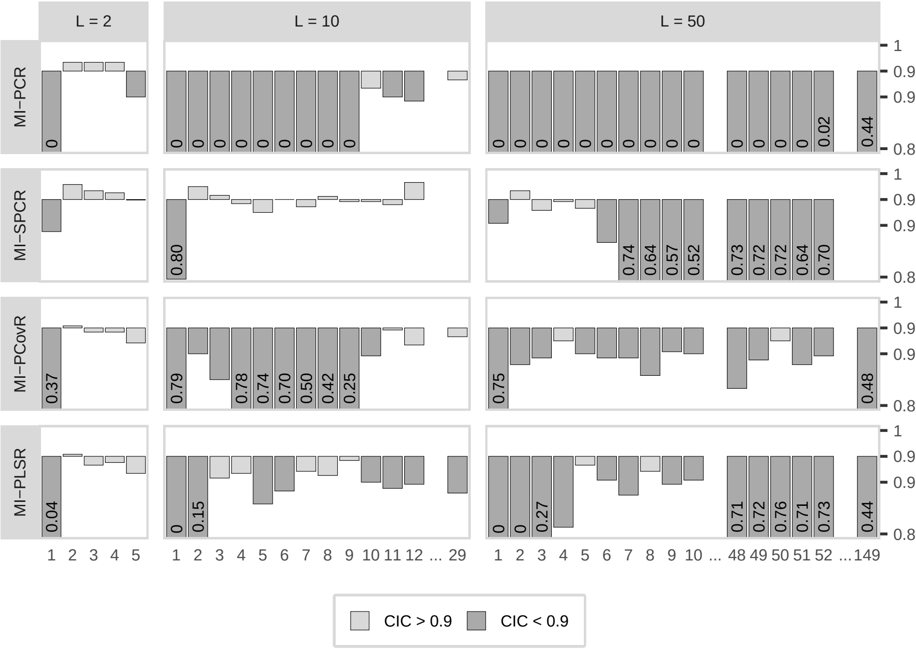


Figure 4.3: The CIC for the estimated correlation coefficient between the first two items imputed is reported (Y-axis) as a function of the number of components (nc) used by the PCA-based imputation methods (X-axis). The plot is divided into a grid where the rows distinguish the results obtained after imputing the data with the four PCA-based methods and the columns distinguish the number of latent variables used to generate the data (L). All results plotted in this figure were obtained on data generated with *mech* = MAR and $pm = 0.5$. For CIC values below 0.8, we reported the precise value within the corresponding bar.

methods resulted in smaller bias, narrower confidence intervals, and closer to nominal coverage than MI-PCR. Considering these results, SDR-based MICE seems to be more appropriate than PCR-based MICE for automatic imputation model specification.

Among the SDR-based methods, MI-SPCR had the best statistical properties. MI-SPCR returned smaller bias and better coverage for a wider range of retained components compared to MI-PCovR. MI-SPCR also achieved a smaller bias than MI-PLSR when retaining fewer components, and it resulted in consistently closer-to-nominal coverages. Based on our results it seems that, at least in the context of the imputation of data with a latent structure, the *discrete* type of supervision employed by MI-SPCR should be preferred to the *continuous* supervision employed by MI-PCovR and MI-PLSR.

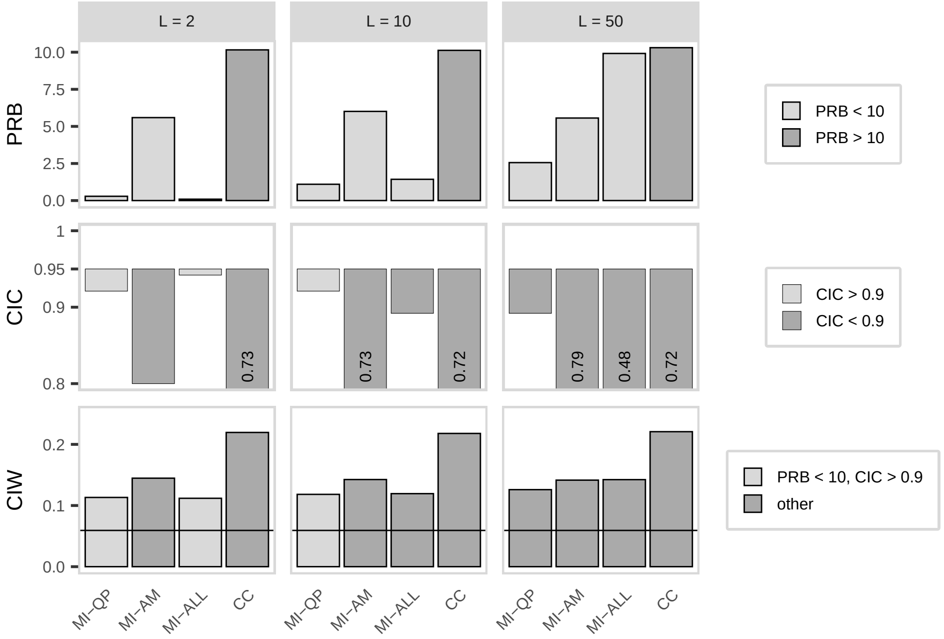


Figure 4.4: PRB, CIC, and CIW for the estimated correlation coefficient between the first two items imputed are reported for the traditional missing data handling methods considered. The black horizontal lines represent the average CIW obtained for the parameter of interest when analyzing the original fully observed data. For CIC values below 0.8, we reported the precise value within the corresponding bar.

4.4.2 Supervision and the number of principal components

Based on our simulation study results, irrespective of which type of supervised dimensionality reduction is used, the implementation of SDR-based methods in MICE should aim for computing a small number of components. Despite this general trend, MI-SPCR and MI-PCovR showed different performances in relation to the different numbers of components retained.

As described in the results section, the bias obtained by MI-SPCR was the smallest when using between 2 and 5 components, independently of the number of latent variables in the data-generating model, and it led to algorithmic failures for large numbers of components. These results can be explained by considering how the number of PC retained influences MI-SPCR. In MI-SPCR, supervision is introduced by pre-screening the columns of possible predictors set to exclude any predictors that are not correlated strongly enough with the variable under imputation. By reducing the number of columns in the predictor set, this supervision reduces the maximum number of components that can be estimated. The inverse constraint also holds. Fixing the number of components puts an upper-bound on the number of variables that can be excluded during the screening process. So, for example, when using 50 components, MI-SPCR must retain, at least, 50 predictors during the screening step, regardless of how weakly some of these variables may associate with any given variable under imputation. Retaining more components forces the threshold values to be smaller and results in keeping more predictors that are less strongly related to the dependent variable, and, by doing so, it limits the advantage of using supervision in the PCA. In conclusion, considering our results and the relationship between the number of components and supervision, we recommend retaining between 2 and 5 components when using MI-SPCR.

In the results section, we noted that in the condition with 10 latent variables, MI-PCovR led to a smaller bias when using 2 components instead of 1, but its bias increased when retaining 3 to 9 components, only to decrease again when retaining 10 or more components. A similar but less extreme trend was also detected in the condition with 50 latent variables, which can be observed in the results dashboard. This fluctuating bias performance can be explained by considering the relationship between the number of components retained and the way we computed α in our simulation study. As described in Equation (4.14), for the same data set, α_{ML} is bigger when the unexplained variance by the components retained is smaller. The more components we retained in our simulation study, the closer the value of α was to 1, and the closer MI-PCovR became to MI-PCR. As a result, MI-PCovR resulted in smaller bias than MI-PCR when retaining the first components, as supervision helped to compute leading components that were important predictors for the imputation task. However, as more components were retained, the effect of supervision started to diminish, which drove the bias closer to that of MI-PCR. After this initial increase, the bias achieved by MI-PCovR dropped when retaining as many components as the number of latent variables, mirroring the drop in bias presented by MI-PCR for the same number of components. As a result, the best performances for MI-PCovR could be achieved by retaining the first components or by retaining a number of components just above the number of latent variables. Because of this fluctuating performance, the optimal range of components to consider when using MI-PCovR is not as clear as for MI-SPCR.

4.5 Limitations and future directions

An important aspect to consider in deciding which version of supervised dimensionality reduction to use with MICE is how flexible these approaches are to deviations from normality of the data. This topic was not covered by our simulation study. MI-SPCR can easily be adapted to impute binary and categorical variables, and the only complication would be in defining a suitable threshold parameter. One option would be to estimate the associations via fit measures derived from simple (multinomial) logistic regression models. Much research has been dedicated to extending PLSR to categorical outcomes (e.g., Chung & Keles, 2010; Ding & Gentleman, 2005), and these approaches could be used in a similar way to the standard PLS implementation we used in this study. The development of PCovR for classification tasks has not received much attention (S. Park et al., 2023, is the only example of which we are aware), but the same approaches used to fit PLS in the generalized linear framework should also apply to PCovR. However, the maximum likelihood estimation of the α parameter can only be done for continuous dependent variables. To impute categorical variables, PCovR would require cross-validation to estimate the value of α, adding to the computational intensity of the procedure.

The set of predictors used to compute the components can also include categorical variables. There are different ways of accommodating these categorical variables when estimating the components, including the naive application of traditional PCA Filmer and Pritchett (2001) and the PCAMIX algorithm (Chavent et al., 2012, 2014; Kiers, 1991) specifically designed for this purpose. Which of these approaches is more appropriate for the predictive task involved in MICE is yet to be tested.

Finally, all of the PCA-based methods considered here (both supervised and unsupervised) entail a high computational load. For every variable and every iteration of the MICE algorithm, complex matrix operations need to be performed to estimate the components. When cross-validating the tuning parameters, the supervised approaches described in this article can increase. Future research should explore possible computational shortcuts to perform supervised dimensionality reduction faster (e.g., Abraham & Inouye, 2014; Halko et al., 2011).

4.6 Conclusions

Based on the simulation study presented here, it can be concluded that adding a supervision element to the classical use of PCR as a univariate imputation method can improve significantly the performance of MI-PCR, especially when the data contain hundreds of variables. Although there is room to assess the performance of these imputation methods in more complex data scenarios, MI-SPCR was particularly effective for the imputation of missing values and seems to be preferable to MI-PCovR and MI-PLSR.

Chapter 5

MICE with generalized supervised principal component regression

Abstract Multivariate imputation by chained equations (MICE) is one of the most popular approaches to address missing values in social science data. Its use by sociologists and other social scientists is still limited by the difficult task of selecting which predictors should be used in the imputation models. In this article, we present the use of a generalized version of supervised principal component regression as a univariate imputation model within the MICE algorithm to reduce the decision-making burden on researchers imputing their data. We show how this approach simplifies the traditional strategy used to select variables for the imputation models while returning imputations of the same quality. We do so by describing an example analysis of the European Values Study.

Results dashboard To run the results dashboard accompanying this chapter install the Shiny app as an R package from the Zenodo permanent repository:

```r
# Install shiny app
devtools::install_url(
    "https://zenodo.org/records/10759084/files/EdoardoCostantini/plotmigspcr-v1.4.zip"
)
```

Then, you can start the app by running this command:

```r
# Start the app
plotmigspcr::start_app()
```

Reproducibility statement The R code used to produce the results reported in this chapter can be found in the permanent repository stored on Zenodo (Costantini, 2024b). The README.md file contains instructions on how to reproduce the results.

A version of this chapter is in preparation for submission.

5.1 Introduction

Researchers in the social, behavioral, and medical sciences often use multidimensional data sets to study how social, psychological, and biological factors shape individual and societal outcomes. Prominent surveys like the World Values Survey and the European Values Study (EVS) are readily available but require addressing the challenge of handling multivariate missing data to unlock their full potential. When missing values are present, the researcher analyzing the data needs to decide how to handle the missingness before they can estimate the analysis model of interest.

Statistical software often uses complete cases when analyzing data with missing values (e.g., R Core Team, 2023; StataCorp, 2023). By focusing the analysis only on the subset of respondents for which all variables are observed, the statistics computed as part of the analysis model can be biased estimates of the population parameter of interest. Furthermore, by reducing the sample size used for the analysis, complete-case analysis (CC) may inflate the standard errors to an undesirable extent.

Rubin (1976) developed Multiple Imputation (MI) as a robust way to address missing values in publicly released survey data. MI generates multiple versions of the original data sets with the missing values replaced by random draws from a predictive distribution based on the observed data. After imputation, an analysis model of interest can be estimated on each differently imputed version of the original data. Finally, the estimates can be pooled to return a single estimate for every parameter. When the missing data mechanism is ignorable (Rubin, 1987), MI allows the use of a larger portion of an incomplete data set compared to CC, while returning unbiased analysis model parameter estimates with valid confidence intervals.

Multivariate imputation by chained equations (MICE, van Buuren & Groothuis-Oudshoorn, 2011) is one of the most flexible ways to implement MI. With MICE, the user does not need to define a single multivariate distribution for all the variables with missing values from which all missing values are drawn. The plausible values are drawn from distributions that are specifically modeled for every variable under imputation, which we refer to as univariate imputation models. This characteristic makes MICE ideal for the imputation of social survey data, usually a collection of numerical, binary, ordinal, and nominal variables for which it is difficult to specify a multivariate distribution.

The specification of the MICE algorithm requires much experience, both in statistical modeling and the research subject matter, and it involves many researcher's degrees of freedom easily inviting analysis error. Using MICE requires making many decisions regarding the imputation models based on the characteristics of the incomplete variables, of the other available variables, of the mechanism generating the missing data, and what type of analysis will be carried out on the imputed data. These decisions are complicated by the lack of clear-cut diagnostic methods to assess whether the main assumption of data being missing-at-random (MAR; Rubin, 1976) holds or whether the algorithm has converged. A variety of guides and software tools have been developed to standardize and simplify these decisions. However, in part due to its complexity and subjectivity, MICE is underutilized in sociological research. Mustillo (2012), Mustillo and Kwon (2015), and Costantini, Lang, Reeskens, and Sijtsma (2023) found that few articles published in high-impact-factor sociological journals use imputation before analyzing the data, and even fewer report the specification details of the imputation procedure they used. In this article, we present the use of generalized supervised principal component regression (GSPCR) in the MICE algorithm to simplify some of the decisions that might be creating obstacles to the use of MICE.

5.1.1 Researcher degrees of freedom in MICE

The specification of the MICE algorithm requires making at least seven decisions (van Buuren & Groothuis-Oudshoorn, 2011). The researcher needs to decide:

1. Whether the MAR assumption is plausible.
2. The number of imputations to generate, which impacts the statistical efficiency of the pooled parameter estimates.
3. The number of iterations, which affects the degree of convergence of the algorithm.
4. The order in which variables with missing values are imputed, which affects the convergence of the MICE algorithm.
5. How to impute variables derived from other variables, such as sum scores, interaction variables, and transformed variables.
6. The type of imputation model to use (e.g., linear, logistic, polytomous regression).
7. Which variables to include as predictors in the imputation models.

Much research has been dedicated to addressing the violation of the MAR assumption (decision 1, Glynn et al., 2000; Heckman, 1976, 1979; N. Little, 2011; R. J. A. Little, 1993). Guidelines exist to help the researchers decide on the number of imputations, the number of iterations, and order of imputations (decisions 2, 3, 4, van Buuren, 2018, pp. 184–189), and how to impute derived variables (decision 5, Eekhout et al., 2014, 2018; Gottschall et al., 2012; Mainzer et al., 2021; van Buuren, 2010). The type of imputation model (decision 6) is mostly determined by the measurement level of the variables under imputation. For example, continuous variables can be imputed using a linear regression model, while binary variables can be imputed using logistic regression. Finally, a variety of decision rules exist to guide researchers in the variable selection for the imputation model (decision 7). However, variable selection remains one of the most delicate decisions in this process, because different variables may produce quite different imputations, and even result in violation of the MAR assumption.

5.1.2 The problem of selecting imputation model predictors

When using MICE, the researcher imputing the data needs to define a set of predictors for the imputation model of every variable under imputation. If the analysis model is known before imputation, the imputation model should at least include all variables that are part of it to avoid biased parameter estimates and invalid inferences (Meng, 1994). Including additional predictors that are not part of the analysis model can improve the efficiency of the parameter estimates (Collins et al., 2001; von Hippel & Lynch, 2013), meaning that all available variables are possible imputation model predictors. Furthermore, leaving out relevant predictors has worse consequences than including useless predictors as the former may result in a violation of the MAR assumption (Collins et al., 2001). As a result, researchers are often recommended to include as many relevant predictors as possible in the imputation models. However, the more predictors are added, the more unwieldy the models become.

With large social surveys, the number of possible predictors may run into the hundreds, and including all of them can easily cause estimation problems in the imputation model, as well as bias and excessive variability of the estimates of interest (e.g., Costantini, Lang, Reeskens, & Sijtsma, 2023; Hardt et al., 2012). The researcher should therefore prioritize variables that are related to the variables under imputation or their missingness. However, this relatedness does not have a precise definition. Measures of bivariate associations between the possible predictors and the variables under imputation or a binary indicator of the missingness are often used to assess whether a possible predictor is worth including. As the number of variables increases, the number of bivariate associations to check increases quadratically and the variable selection task becomes cumbersome. Furthermore, if a variable included as a predictor also has missing values, this variable needs to be imputed too and the researcher needs to decide which predictors to include in this new imputation model as well, further increasing the number of decisions.

Due to the complexity of the selection of the predictors, most implementations of MICE provide some automatic decision strategy. One approach is to define a correlation threshold as a general decision rule. One can compute a correlation matrix for the data at hand and include the variables that correlate higher than a predefined correlation value. This approach is the basis for the model-building strategy implemented by the popular R package *mice* (van Buuren & Groothuis-Oudshoorn, 2011) in the `quickpred()` function. Another approach is to build every imputation model with a forward stepwise regression (Efroymson, 1966) where every variable under imputation is regressed on increasingly more complex models including more predictors than the previous model until the increase in explained variance becomes negligible. This approach is implemented in the stand-alone Imputation and Variance Estimation Software (IVEware, Raghunathan et al., 2002).

These strategies offer a fast, data-driven mechanism to specify the imputation models, but they also have their drawbacks. The correlation-threshold strategy can select too many predictors. The step-forward strategy can select different predictors depending on the order in which predictors are checked. Furthermore, both strategies are difficult to apply when categorical variables are part of the data.

Recent research in the field of MI has focused on integrating high-dimensional model-building strategies in the MICE algorithm to handle large sets of possible predictors. For example, Deng et al. (2016) proposed the use of lasso regularization as a tool to perform data-driven variable selection, while others have experimented with non-parametric prediction models (decision trees, Burgette & Reiter, 2010), and dimensionality reduction techniques (Howard et al., 2015; Robitzsch et al., 2016). Costantini, Lang, Reeskens, and Sijtsma (2023) compared the use of these and other high-dimensional techniques as imputation models with more traditional model-building strategies and found the use of principal component analysis (PCA; Jolliffe, 2002) to perform well and to show the greatest potential to become a simple-to-use and effective strategy for building imputation models. Costantini et al. (2024) found that the use of principal component regression (PCR; Jolliffe, 2002, pp. 168–183) as a univariate imputation model outperformed other combinations of PCA with the MICE algorithm.

A limitation of PCR is that the principal components (PCs) that represent the largest variance in the independent variables might be poor predictors of a dependent variable of interest (Bair et al., 2006; de Jong & Kiers, 1992). In the context of MICE, this could mean that poor imputations are obtained when the PCs used as predictors are computed without considering the variables under imputation. A variety of dimensionality reduction approaches have been proposed to obtain PCs that represent the possible predictors well and that are also good at predicting the dependent variable (Bair et al., 2006; de Jong & Kiers, 1992; Wold, 1975). Costantini, Lang, and Sijtsma (2023) identified supervised principal component regression (SPCR; Bair et al., 2006) as the dimensionality reduction technique that is most effective when used as a univariate imputation model in MICE.

5.1.3 Research goals

In this article, we demonstrate the use of GSPCR, an extension of SPCR for data that are a mix of continuous and categorical variables, as a univariate imputation model for MICE to generate imputations for a large survey data set. We refer to this approach as MI-GSPCR. A researcher using MI-GSPCR does not need to select predictors for the imputation models (decision 7) because all of the possible predictors are processed by GSPCR. By considering all potential predictors, MI-GSPCR not only reduces the burden of decision-making on the researcher but also stabilizes estimation and decreases the chances of ignoring important predictors.

We present an example imputation and analysis of the EVS (2020a) data using both MI-GSPCR and a specification of MICE where predictors for the imputation models are selected as recommended by current best practices. We compare the imputations generated by the methods and the pooled parameter estimates for an example analysis model. The article is also accompanied by an R tutorial providing a guide for readers interested in trying the approach with their data (see Appendix B).

5.2 Imputation methods and algorithms

5.2.1 Multivariate imputation by chained equations

Like any other MI procedure, MICE takes as its input a data set with missing values and it returns multiple versions of this original data set with the missing values replaced by plausible values. A researcher performing MI is usually interested in estimating one or more analysis models on the data. An analysis model can be estimated on each of the multiple versions of the data, resulting in multiple estimates for the same parameters, which are then combined (or *pooled*) according to standard rules defined by Rubin (1987).

The MICE algorithm begins by replacing the missing values on all the incomplete variables with starting values typically randomly sampled from the observed values of the variables under imputation. Then, MICE consecutively replaces the missing values for each separate variable under imputation, often starting with the variable with the fewest missing values, in a two-step procedure:

1. First, MICE samples the parameter values for the imputation model predicting the variable under imputation from a posterior distribution appropriately defined based on the likelihood function and priors.

2. Then, MICE samples imputations for the missing values by sampling from a predictive distribution based on the samples in step 1.

After completing these steps, the algorithm moves to the next incomplete variable and replaces the missing data by repeating the same two steps. When the missing values in this second variable have been replaced with new values, the algorithm moves to the next incomplete variable and repeats the same steps. Once the algorithm has sampled new values for all the incomplete variables, the second iteration starts. New values are sampled to replace the missing values in the first variable, but now all of the predictors with missing values have been updated with values that are more plausible than the starting values. The algorithm continues by cycling again through all the incomplete variables for I iterations. Because each variable's imputation depends on the imputed values of other variables, and because this dependency propagates through the iterations, we refer to the collection of imputed values generated through the iterations as a stream of imputed values. When the algorithm reaches the Ith iteration, the last values in the stream are considered to be one set of imputations. Multiple sets of imputations can be sampled by generating multiple independent streams.

5.2.2 Choosing predictors for the imputation models

A researcher using MICE needs to select an appropriate set of predictors for the imputation model of every variable under imputation (decision 7). This is usually done by considering how helpful every variable in the data could be for the imputation procedure. To explain how the predictors are selected, let us consider an example. Imagine a researcher who wants to analyze a data set composed of three ordinal

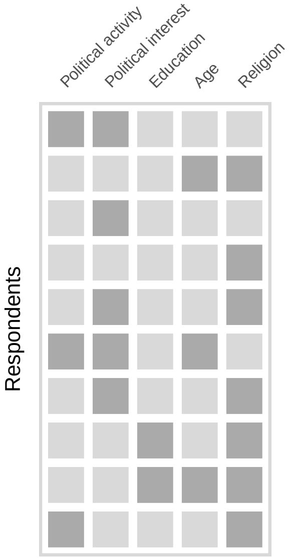


Figure 5.1: Missing data pattern in an example survey data. Every square represents the answer given to a question by a respondent. The light gray squares represent observed values, while the dark gray squares represent missing values.

variables (*Political activity*, *Political interest*, and *Education*), one continuous variable (*Age*), and one nominal variable (*Religion*). The analysis model the researcher is interested in is a linear regression model with *Political activity* as a dependent variable and *Political interest* and *Education* as independent variables. The data are characterized by a general missing data pattern, as represented in Figure 5.1. Analyzing only the complete cases would result in using only one-fifth of the original data and in biasing parameter estimates. Therefore, the researcher decides to generate multiple imputations for all variables in the analysis model.

The researcher starts by considering which predictors to use in the imputation model for *Political activity*. *Political interest* and *Education* are both part of the analysis model, so they should be included as predictors of *Political activity*, as leaving them out might produce bias in the analysis model parameter estimates. Two additional variables in the data set, *Age* and *Religion*, are potential predictors for the imputation models, even though they are not part of the analysis model.

These potential predictors can be helpful for imputation if they are associated with the respondents' scores on *Political activity*, with the probability of having a missing value on *Political activity*, or with both. To assess the suitability of *Age* as a predictor for *Political activity*, the researcher must evaluate the strength of their association, for example by calculating the correlation between *Age* and *Political activity*. If the correlation is strong, including *Age* will reduce the uncertainty of the imputations. The researcher should also consider whether *Age* helps predict the probability of having missing values on *Political activity*. This can be established on substantive grounds (i.e., arguing why different age groups are less likely to respond to questions regarding their political activities), or by calculating an association measure between *Age* and a binary response indicator recording which values of *Political activity* are observed and which values are missing. If *Age* is associated with both *Political activity* and its response indicator, then excluding it from the imputation model would result in a violation of the MAR assumption, hence *Age* must be included as a predictor. Once the researcher has decided whether to include *Age*, they can apply the same logic to assess whether *Religion* is a valuable predictor for *Political activity*.

After choosing the predictors for the imputation model of *Political activity*, the researcher can use the same reasoning to define the predictors of the imputation models of *Political interest* and *Education*. It is common for predictors used in the imputation models to also have missing values, which requires specifying predictors for their imputation models too, easily leading to a situation where an imputation model needs to be specified for all variables in the data. In our example, if either *Age*, *Religion*, or both, have been identified as predictors for any imputation model, the researcher also needs to define an imputation model for these variables. Evaluating whether *Age* and *Religion* should be used as predictors in each other's imputation models would require the additional computation of three association measures: the associations between *Age* and *Religion*, *Age* and the response indicator for *Religion*, and the response indicator for *Age* and *Religion*.

When the number of possible predictors becomes large, the number of association measures to check increases almost quadratically. For example, in a data set with 100 possible predictors, there are at least $100 * (100 - 1)/2 = 4,950$ variable-to-variable associations and $100 * (100 - 1) = 9,900$ variable-to-response-indicator associations to consider. With hundreds of possible predictors, manually checking the strength of associations is infeasible and more data-driven approaches are needed. One approach is to define a minimum strength of association and include as predictors all variables that are more strongly associated with the variables under imputation than that threshold.

This threshold approach is fast and easy to implement but it requires deciding on an association measure that is comparable for every type of variable involved. Different measurement levels require different association measures which are not directly comparable. For example, the same numerical values of Spearman Rank correlation oefficient (Spearman, 1904) between two ordinal variables and the Cramér's V (Cramér, 1946) coefficient between an ordinal and nominal variable do not represent the same association strength, thus hampering model selection for mixed data types using association thresholds.

Costantini, Lang, Reeskens, and Sijtsma (2023) showed that naively applying the threshold approach to social surveys with hundreds of variables can lead to a large bias in the estimates of the parameters in the analysis model. The researcher needs to carefully select a threshold that allows the inclusion of enough important predictors without selecting so many variables as to create estimation problems. To keep the number of predictors contained, the researcher might opt to use only the 10 or 20 most strongly associated variables as predictors for every variable under imputation. They might also decide to use a ridge penalty to avoid computational singularity and perform collinearity checks to exclude strongly correlated predictors. These measures can help estimation of the imputation models but they introduce more researcher's degrees of freedom and increase the decision-making burden of selecting predictors for the imputation model.

5.2.3 Simplifying the specification of imputation models with GSPCR

Costantini, Lang, and Sijtsma (2023) proposed the use of SPCR as an imputation model in MICE to simplify the choice of the imputation model predictors (MI-SPCR). In this approach, SPCR summarizes all the important predictors with a few relevant PCs and uses them as predictors in the imputation models. The researcher using MI-SPCR does not need to select predictors, define the threshold value or choose a maximum number of predictors. Furthermore, they do not need to specify the ridge penalty or perform collinearity checks as PCs are linearly independent by definition. Because all predictors are used to compute the PCs, all predictors with missing values need to be imputed. However, the researcher can use SPCR to define all imputation models, which reduces the burden of specifying imputation models for predictors of predictors that characterizes the strategy described in Section 5.2.2.

SPCR is a dimensionality reduction approach to regression where a dependent variable is regressed onto a set of PCs computed on an original set of independent variables. Compared to traditional PCR, SPCR prioritizes the computation of relevant PCs for the prediction of the dependent variable by restricting attention to independent variables that are highly correlated with the dependent variable. As presented by Hastie et al. (2017, pp. 674–678), SPCR can be described by the following steps:

1. Estimate all the simple linear regression models regressing the dependent variable onto each independent variable separately.
2. For each value of a threshold ρ_l from the list $0 \leq \rho_1 < \rho_2 < \cdots < \rho_L$:
    a) Compute the first Q PCs of the data including only the independent variables with regression coefficients higher than ρ_l.
    b) Regress the dependent variable onto these PCs.
3. Pick the value of ρ_l and Q by cross-validation.

In the MI-SPCR approach, SPCR is estimated by using the observed values on the variable under imputation as the dependent variable and the corresponding values on all the possible predictors as independent variables. The SPCR model is then used to generate predictions for the missing values on the variable under imputation and noise is added to include uncertainty regarding the imputations. Uncertainty regarding the parameters of the imputation model is incorporated by taking a bootstrap sample of the data before estimating SPCR, a standard way of obtaining proper multiple imputations (van Buuren, 2018, pp. 67-69). These steps are repeated for every variable under imputation, at every iteration of MICE. A detailed description of this imputation algorithm can be found in Costantini, Lang, and Sijtsma (2023).

The original SPCR was developed to summarize gene expression data, restricting its applicability to continuous data. Consequently, MI-SPCR can generate imputations only for continuous data. To overcome this limitation, and allow for the imputation of the different types of variables that are present in survey data, we developed an extension to SPCR that uses generalized linear modeling framework (GLM; Nelder & Wedderburn, 1972) to establish the relationship between the dependent variable and the PCs, allowing for continuous, binary, ordinal, and categorical dependent variables. Furthermore, we allowed the inclusion of categorical predictors along with numeric predictors by computing the PC scores through a combination of PCA and multiple correspondence analysis (MCA; Tenenhaus & Young, 1985) known as PCAmix (Chavent et al., 2012, 2014). We refer to this generalization of SPCR as GSPCR. In Appendix A, we explain the GSPCR approach in greater detail.

We implemented an algorithm to estimate GSPCR in the R package *gspcr* (Costantini, 2023b) and we developed additional functions for the *mice* R package to demonstrate the potential use of GSPCR as a univariate imputation method to generate multiple imputations for data with variables of any measurement level. We refer to this use of MICE as MI-GSPCR. Using MI-GSPCR is as simple as specifying a different method argument in the `mice()` function, as shown in the tutorial in Appendix B. Because GSPCR plugs into the MICE algorithm as just another univariate imputation model, convergence checks, imputation of derived variables, and sensitivity analysis to address potential violations of the MAR assumption can be performed as with any other MICE specifications.

5.3 Multiple imputation of the European Values Study data

To demonstrate how MI-GSPCR can be used in practice, we imputed real survey data and carried out an example analysis. In this section, we describe the data, the analysis model, the missing data problem, and how MI-GSPCR can be specified. We report imputation diagnostics and the analysis model parameter estimates resulting from MI-GSPCR. We compare these results with those obtained by imputing the data with a traditional specification of MICE and by only analyzing the complete cases.

5.3.1 Data structure and processing

The EVS is a cross-national and longitudinal social survey on relevant moral, political and social values and attitudes. Over the years, it has been an important source of information to document social (values) change (Inglehart, 1997, Luijkx et al., 2021, pp. 330–346). The majority of the variables are survey items with ordinal scoring, but there are also many unordered categorical and binary variables and a few numerical variables. The ordinal items usually have between 3 and 10 categories and their distribution can be symmetrical, skewed, and even multi-modal. The variety of distributions and measurement levels make the EVS data a good example of the type of data sociologists and other social scientists regularly use in their analysis. EVS data are freely available to researchers.

We used the third prerelease of the 2017 wave of the EVS data (EVS, 2020a) which contains responses to 466 variables from a sample of 56,491 individuals. These data contain mode, administrative, and questionnaire variables and weights. Mode variables record how the interview took place, whether the interview took place in person or online, and whether the individual was assigned to any planned missingness design. Administrative variables record meta-data regarding the survey (e.g., year of administration, DOIs, interview language) and the respondent (e.g., case ID). We kept all mode and administrative variables that provided potentially useful information for either the analysis or the imputation procedure (e.g., the mode of the interview might be related to missing data patterns) and discarded the rest (e.g., participants' identification codes are useless for the analysis). Weights are variables meant to adjust some socio-demographic characteristics in the sample to the distribution of the target population in each country. We ignored weights because the focus of this article is on imputation, not generalization of the results to a population. We kept 248 variables in total (48 binary, 38 nominal, 158 ordinal, 2 continuous, and 2 count variables). The R code used for the data cleaning can be found in the permanent code repository associated with this article Costantini (2024a).

5.3.2 Substantive analysis model

We defined a plausible analysis model based on a review of articles cited in the repository of publications using EVS data (EVS, 2020b). The model we selected is inspired by Immerzeel et al. (2015), who tried to explain gender differences in voting behaviors by gender differences in socioeconomic characteristics and radical right attitudes. One of the research questions in Immerzeel et al. (2015) was: 'To what extent is there a gender gap in Western European radical right voting, and, if there is a gender gap, to what extent can this gap be explained by gender differences in socioeconomic characteristics and radical right attitudes?'

For this study, we slightly simplified the analysis models. The voting tendency of the respondents was operationalized by the question 'Which (political) party appeals to you most?' with country-specific response options located on the left-to-right political spectrum. This variable had around 500 categories, representing all the political parties in the 12 countries of interest. Immerzeel et al. (2015) defined a binary dependent variable by manually coding whether the party respondents chose could be considered a right-wing party or not. Creating this indicator in the presence of missing values would require first imputing the party choice and then performing the manual coding. The imputation of a nominal variable with this many categories is impractical and outside the scope of this article. Therefore, we operationalized voting tendencies with a numerical left-to-right harmonized variable (v275_LR) provided in the EVS data instead of a manually coded binary indicator. As a consequence, our analysis model was defined as a multiple linear regression model instead of a logistic regression model.

Following Immerzeel et al. (2015), we used the following independent variables:

- *Gender*, operationalized by the response to question 'Are you a man or a woman?' (v225).

- *Occupational class*, coded according to the Erikson et al. (1979) class scheme.
- *Nativist attitudes*, operationalized by attitudes toward immigrants and immigration as represented by the average of respondents' agreement with three statements: 'immigrants take jobs away from natives' (v185), 'immigrants increase crime problems' (v186), and 'immigrants are a strain on welfare system' (v187). Agreement with these statements was scored on a scale from 1 to 10.
- *Authoritarian attitudes*, operationalized by two predictors: item v145, which asked to express the importance assigned to having a strong leader on a scale from 1 (very good) to 4 (very bad), and item v110, which asked the respondent to choose the most important aspect between 'maintaining order in nation', having 'more say in important government decisions', 'fighting rising prices', and 'protecting freedom of speech'.
- *Political interest*, operationalized by question 'How interested would you say you are in politics?' (v97) with possible scores from 1 (very interested) to 4 (not interested at all).
- *Political activity* of respondents, operationalized by the average of their willingness to participate in a petition (v98), boycott (v99), demonstration (v100), and strike (v101), scored as 1 (have done), 2 (might do), and 3 (would never do).
- *Age* of the respondents when interviewed in 2017 (age_r3), coded into seven age groups: 18–24, 25–34, 35–44, 45–54, 55–64, 65–74, and 75+ years old.
- Highest level of *education* completed as classified by the International Standard Classification of Education (v243_ISCED_1). Following Immerzeel et al. (2015) we recoded this variable to have three categories: primary, secondary, and tertiary education.
- *Marital status* of the respondent (v234), operationalised in three unordered categories 'having a partner', 'never had a partner', and 'had a partner' (divorced, separated, and widowed).
- Degree of *urbanization* of the respondents' place of residence (v276_r), recorded as either one of '< 5000', '$5001–20,000$', '$20,001–10,0000$', and '$> 100,000$' inhabitants.
- Religious *denomination* (v52), coded into 'non-religious', 'Christian', or 'other religions'.

- Religious *attendance*, as operationalised by the response to ‘Apart from weddings, funerals, and christenings, about how often do you attend religious services these days?’ (v54). Respondents could answer on a scale from 1 (never) to 7 (weekly) but the variable was recoded to have three categories: never, sometimes (including ‘less than once a year’, ‘only on specific holidays’), often (‘once a month’ or more).

- *Country* where the interview took place. Following the research question, we analyzed the subsample of 15,109 observations interviewed in the 12 Western European countries[22] considered by Immerzeel et al. (2015).

5.3.3 Missing data problem

Because all 248 variables are used as predictors by MI-GSPCR, we had to impute all of the variables with missing values. Of the 248 variables, only *country* and *mode* were fully observed. In the remaining 246 variables, the proportion of missing cases (pm) ranged from 0.02% to 40%. The respondents’ voting tendency, the dependent variable in the analysis model, was 30% missing. The analysis model predictor with the highest proportion of missing values was the degree of *urbanization* ($pm = 0.10$), which was mostly due to the difficulty in gathering this information in the Netherlands, one of the countries of interest. Predictors concerning attitudes had missing data rates below 10%. For example, the items measuring authoritarian attitude had around 5% cases missing. Demographic predictors like age and education level had fewer than 1% cases missing. Using only complete cases (1,579) for data analysis would imply discarding 90% of the sample, a loss of information that would decrease precision and produce biased results.

5.3.4 Imputation strategies

We generated multiple imputations for the 246 variables with missing values using MI-GSCPR. All available variables in the EVS data were included as predictors in the imputation model of every variable under imputation. The GSPCR algorithm used the Cox and Snell pseudo R^2 (R^2_{CS}; Nagelkerke, 1991, see Appendix A for more details) to determine the association strength between the variables to be imputed and the possible predictors and BIC as a fit measure for the cross-validation procedure used to choose the threshold R^2_{CS} value and the number of PCs returning the best

---

[22] Austria, Belgium, Denmark, Finland, France, Germany, France, Italy, Luxembourg, the Netherlands, Norway, and Switzerland.

model fit. The possible values of R^2_{CS} were set to the 0.1 intervals between 0 and 1, while the possible numbers of PCs were $1, 2, 3, 4$, and 5, based on recommendations provided by Bair et al. (2006) and the findings of Costantini, Lang, and Sijtsma (2023).

We also generated multiple imputations with a run of MICE using standard imputation models specified by carefully selecting predictors following the logic described in Section 5.2.2. To select the predictors, we computed the bivariate associations between every variable (variable-to-variable), and the association of every variable with every response indicator (variable-to-response). For both the variable-to-variable and the variable-to-response associations, we used R^2_{CS} as an association measure because it allows for a standardized comparison of the association between variables of any measurement level. For every variable pair, we defined a single association measure ρ_{max} as the higher value between the variable-to-variable and the variable-to-response association. That is, the ρ_{max} between *Political activity* and *Age* was either equal to the R^2_{CS} between *Political activity* and *Age* or the R^2_{CS} between the response indicator for *Political activity* and *Age*, whichever one was higher.

Figure 5.2 shows the 30 best possible predictors for four incomplete variables of different measurement levels ordered based on ρ_{max}. These bar plots show that after approximately 15 variables it became difficult to distinguish the possible predictors based on the strength of association with the targets of imputation or the response indicators. As a result, we selected the first 15 variables with the highest ρ_{max} as predictors for the imputation models of every variable under imputation. We also made sure that all analysis model variables were included as predictors in the imputation model of every analysis model variable with missing values. Finally, we defined a fixed ridge penalty equal to 0.00001 to stabilize estimation and we excluded predictors correlating higher than 0.99, which are default choices in the *mice* package. Because this approach implements the standard logic described in Section 5.2.2 to select the predictors for the imputation models in MICE, we refer to it as MI-Standard.

For both MI-GSPCR and MI-Standard:

- We used the `mice()` function from the *mice* package to perform imputation;
- We set the initial number of iterations to 20, which is four times the number of default iterations in the *mice* package.
- We generated five imputed data sets, the default value in the *mice* package;

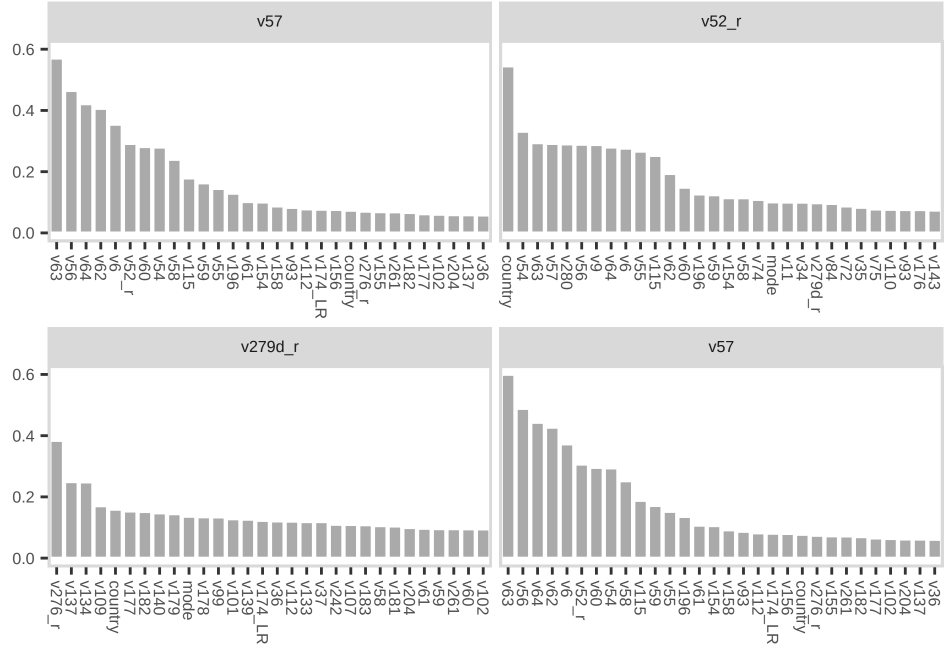


Figure 5.2: The 30 possible predictors with highest ρ_{max} (X-axis) for four example variables under imputation (v57, v52_r v279d_r, v239_r), ordered based on ρ_{max} (Y-axis).

- We treated derived variables with the impute-then-transform strategy (von Hippel, 2009);
- Variables were imputed in the order they appeared in the data set, from left to right.
- We used logistic regression, polytomous logistic regression, and predictive mean matching (PMM, R. J. A. Little, 1988) to impute binary, nominal, and ordinal data, respectively. We also used PMM to impute count data, but we imputed ordinal variables with at least five response categories and symmetrical distributions with the normal linear regression model. Appendix B.2 provides more details on how these methods were implemented for MI-GSPCR.
- We operated under the MAR assumption. When analyzing social science data, it is common to start by assuming MAR and then using sensitivity analyses to assess the impact of this assumption on the imputations. Because the violation of the MAR assumption is not the focus of this article, we did not perform

these sensitivity analyses here, but if required they could be performed with MI-GSPCR and MI-Standard in the same manner.

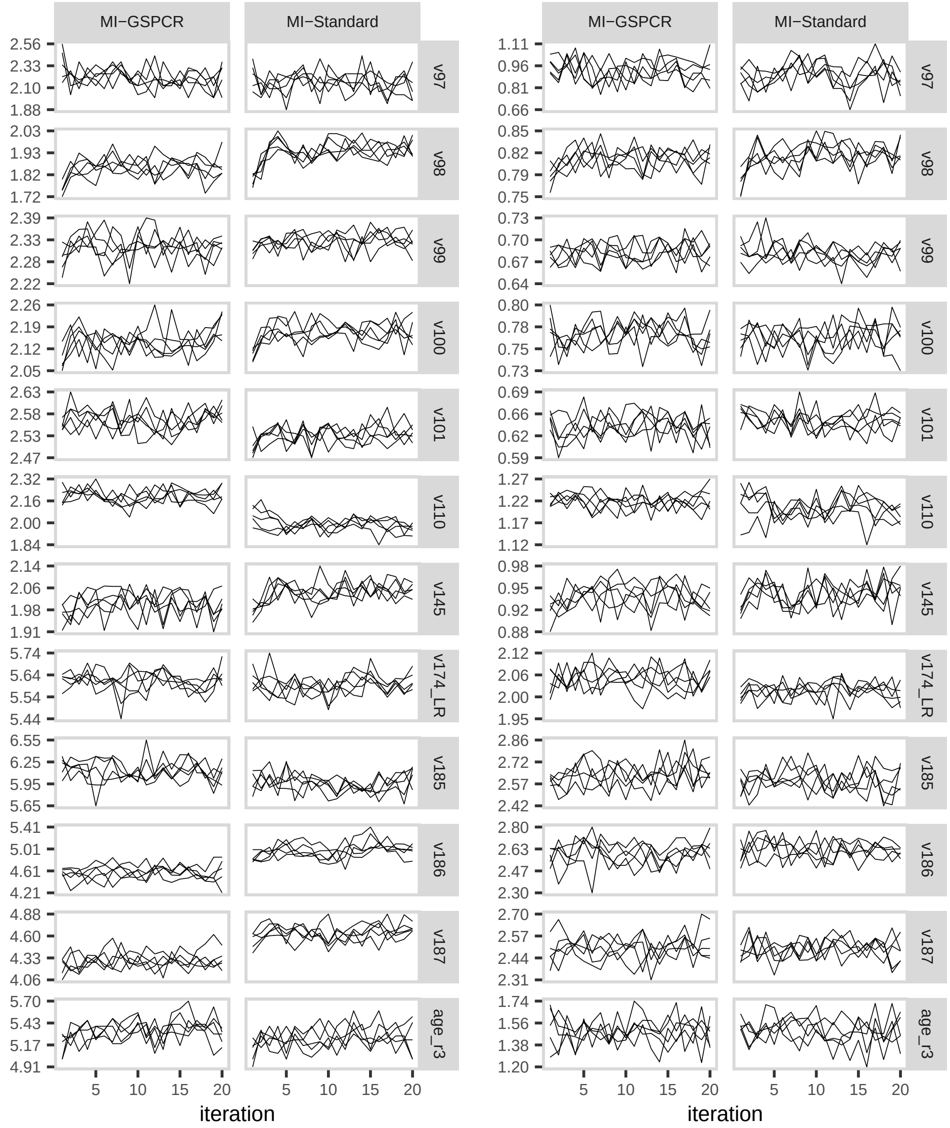
5.3.5 Imputation results

After the imputation, we performed diagnostic checks to evaluate the convergence of the algorithm and the quality of the imputations. We also estimated the analysis model described in Section 5.3.2 and pooled the parameter estimates. In what follows we compare these results for MI-GSPCR and MI-Standard.

This study is accompanied by an interactive results dashboard packaged as an R Shiny app (Costantini, 2023f). In this dashboard, the user can review the diagnostic measures for all impute variables and plot additional comparison measures that were not reported here for brevity. The software can be downloaded and installed as an R package. A user manual is included as a README file in the folder accessible through the DOI provided in the citation. We encourage the interested reader to use this tool while reading the following sections.

5.3.5.1 Convergence

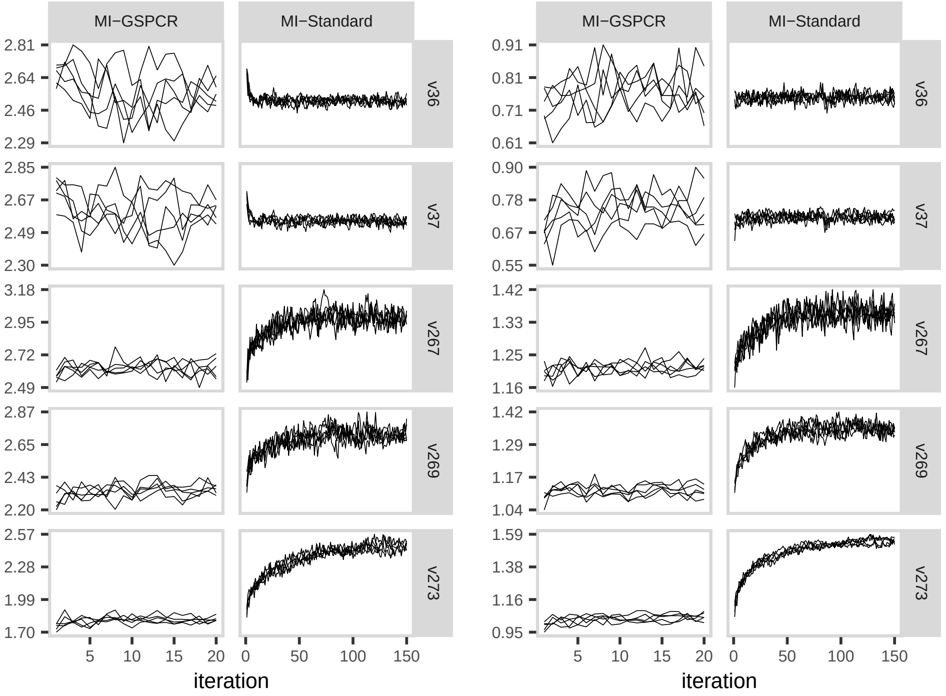
Figure 5.3 shows the mean and standard deviation of the imputed values for the ordinal variables in the analysis model against the iteration number for five streams of imputations obtained with MI-GSPCR and MI-Standard. The lines should freely intermingle without showing any definite trend. When looking at the analysis model variables, the convergence trends for MI-GSPCR and MI-Standard did not indicate any problematic convergence. However, for MI-Standard, some variables that were not part of the analysis model showed signs of non-convergence. We observed that the imputations of variables v36 and v37 generated by MI-Standard showed a downward trend in the convergence plot for the first 20 iterations. Similarly, the imputations of variables v267 to v274 generated by MI-Standard showed an upward trend in the mean and standard deviations of the imputed values over the initial 20 iterations. Therefore, we decided to run additional iterations of MI-Standard. Figure 5.4 shows that approximately 150 iterations were needed to achieve convergence of the imputations with MI-Standard, while MI-GSPCR converged within the first 20 iterations. The trace plots for all variables are available in an interactive results dashboard that was developed to accompany this article (Costantini, 2023f).



(a) Average imputed values.

(b) Standard deviation of the imputed values.

Figure 5.3: Trace plots for imputations on the analysis model variables showing the trends in the imputed values average values and standard deviations for the different imputation streams across iterations of the MICE algorithm.



(a) Average imputed values.

(b) Standard deviation of the imputed values.

Figure 5.4: Trace plots for imputations with slow convergence showing the trends in the imputed values average values and standard deviations for the different imputation streams across iterations of the MICE algorithm.

5.3.5.2 Density plots

The distribution characteristics of the imputations were examined by comparing the distributions of the observed and imputed values. We examined the density plots for all imputed variables and found that both MI-GSPCR and MI-Standard returned imputations that were distributed similarly to the observed data for most variables, even when the data had non-standard distributions (e.g., skewed or multi-modal). For brevity, we report two example density plots comparing these distributions in Figure 5.5.

5.3.5.3 Analysis model parameter estimation

In Figure 5.6, we report the absolute parameter estimates of the analysis model, the within-imputation variance of the estimates ($\bar{U}$), and the between-imputation vari-

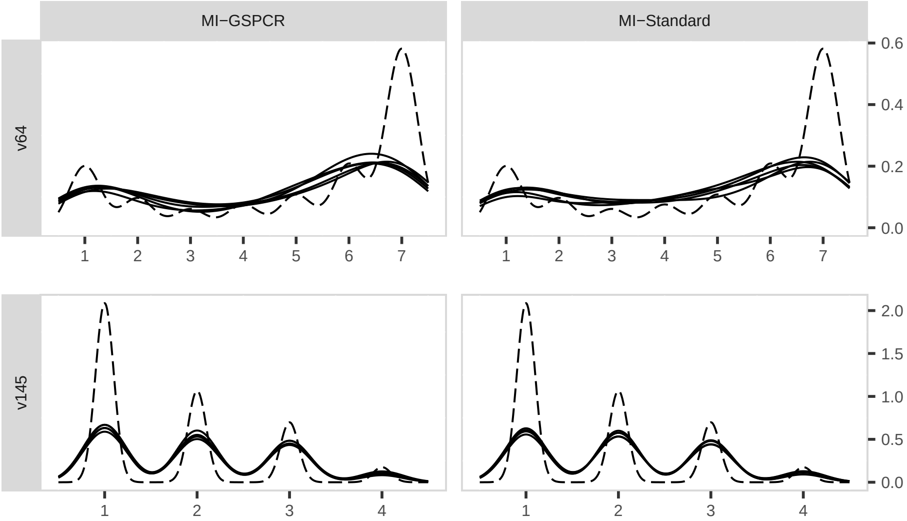


Figure 5.5: Density plots for the observed versus imputed values on two non-normal variables in the data set. The dashed line represents the density of the observed data. The solid lines represent the densities of the imputed values at the end of the different streams.

ance (B) obtained after combining the multiple estimates obtained with MI-GSPCR and MI-Standard. In general, the two methods produced similar parameter estimates. Some differences were found in the estimates for the regression coefficients of dummy variables representing *Urbanization*, *Denomination*, and *Country*, but neither method returned a consistently lower or higher estimate than the other. As for the estimates' variability, $\bar{U}$ was indistinguishable between the two methods, but MI-GSPCR resulted in smaller B than MI-Standard for some coefficients, while MI-Standard resulted in smaller B for others. This was particularly noticeable for the regression coefficients of the dummy variables representing *Occupational class* and *Country*.

In Figure 5.7, we report the pooled parameter estimates and pooled standard errors obtained by MI-GSPCR and MI-Standard along with the results obtained by estimating the analysis model on the complete cases. The pooled standard errors for MI-GSPCR and MI-Standard were computed as a weighted sum of B and $\bar{U}$ as described by Rubin (1987, p. 77). In general, the estimates and standard errors of the parameters obtained from MI-GSPCR and MI-Standard were relatively close compared to CC. CC resulted in parameter estimates that were twice or half the size of the parameter estimates obtained by MI-GSPCR and MI-Standard. Furthermore, the standard errors obtained with CC were more than twice as large as those obtained by MI-GSPCR and MI-Standard.

5.3.5.4 Computational intensity and imputation time

The 20 iterations used for MI-GSPCR were sufficient to achieve convergence of the imputation streams, while MI-Standard needed 150 iterations to reach convergence. With 150 iterations, MI-Standard took almost a day to run (20 hours), while the 20 iterations of MI-GSPCR took approximately a week. These run times depend on the computer specifications, but the current implementation of MI-GSPCR has a higher computational demand than MI-Standard.

5.4 Discussion

When imputing large survey data with MICE, defining good imputation models implies many researcher degrees of freedom and requires great statistical modeling expertise. In particular, when selecting the predictors for the imputation models in MICE, researchers need to make decisions about which association measures to use for variable selection, which threshold identifies important predictors, how many predictors to keep, the value of the ridge penalty to stabilize the computation, and how to perform collinearity checks. MI-GSPCR imposes a lower decision-making burden on researchers by requiring only the specification of the type of imputation model. While it is true that the GSPCR model inside MI-GSPCR can be specified in different ways, the specification we described in Section 5.3.4 applies to a wide range of real data scenarios. For example, the Cox-Snell R-squared and the BIC were chosen to define the association measure and the fit measure precisely because they can be used with the different types of variables often recorded in surveys. Similarly, we chose the range of PCs based on literature showing that when using SPCR it is usually unnecessary to use more than five PCs.

Another important feature of MI-GSPCR is that it does not disrupt established MI frameworks to check the quality of imputations. All diagnostics tools that can be used for MICE can also be used for MI-GSPCR. When analyzing trace plots we noticed that for certain variables MI-Standard required more iterations than MI-GSPCR to reach convergence. Different imputation model choices could result in different convergence rates, but this result does suggest that MI-GSPCR can converge within fewer iterations than MI-Standard when imputing large surveys. However, increasing

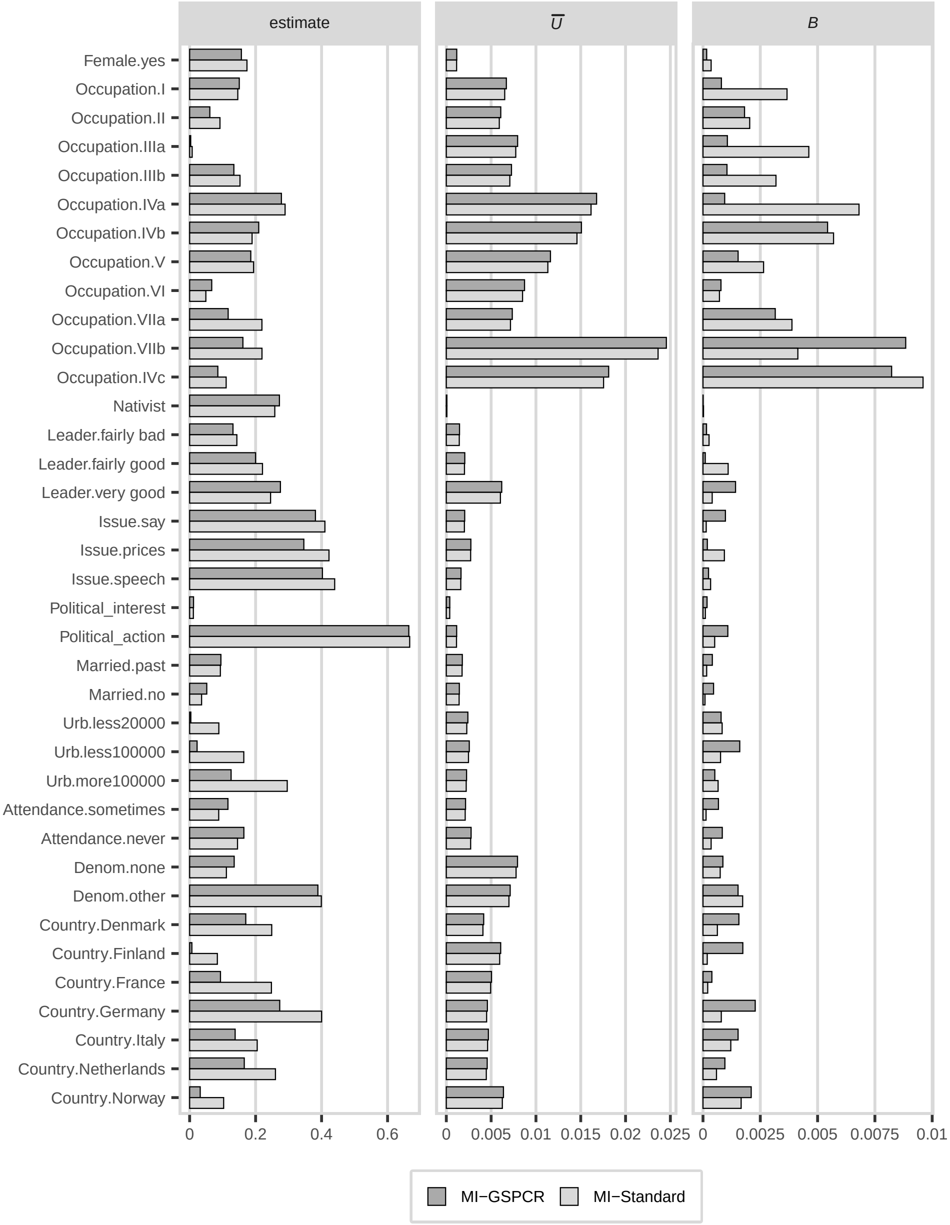


Figure 5.6: Regression coefficient estimates (in absolute value), within-stream variance ($\bar{U}$), and between-stream variance (B) of the estimates obtained after using MI-GSPCR and MI-Standard.

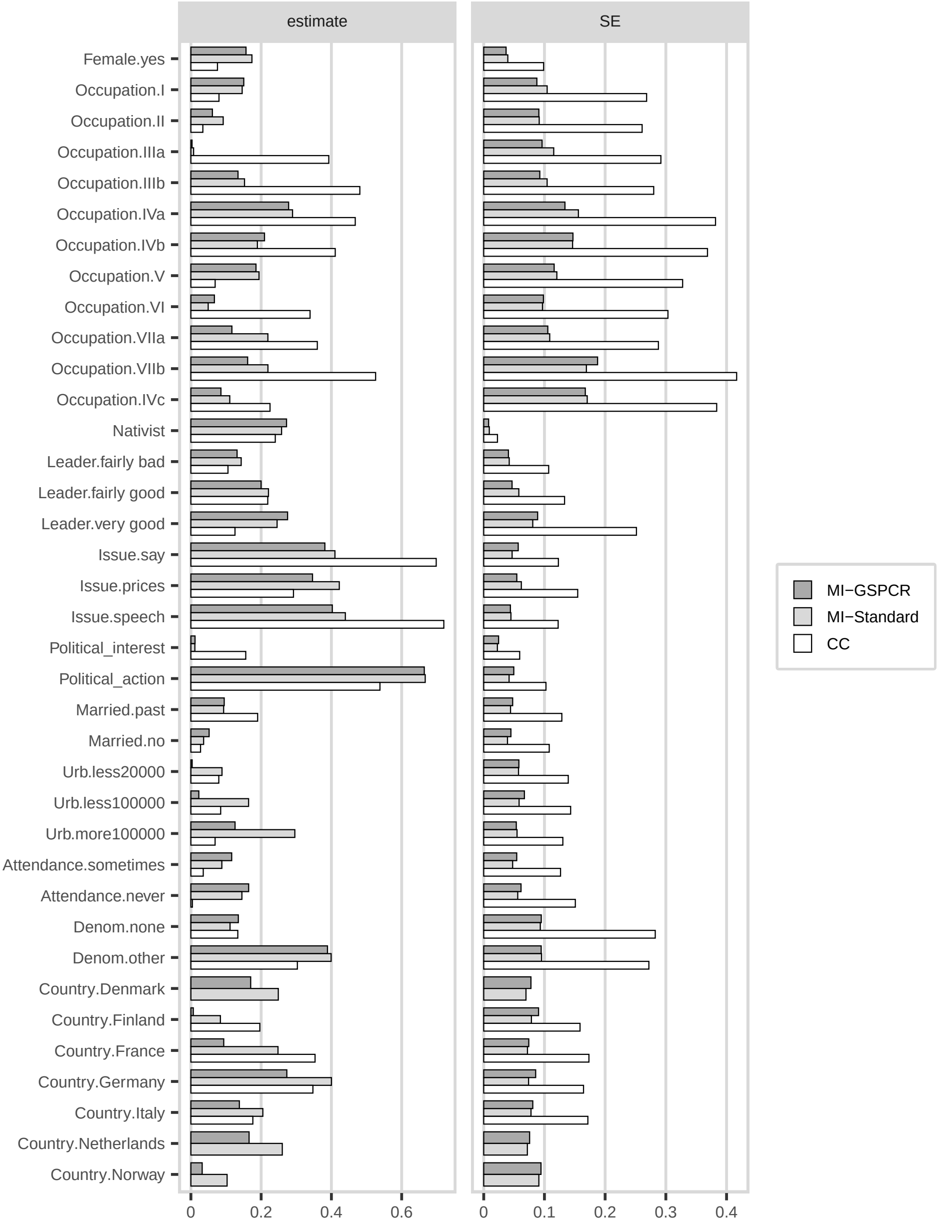


Figure 5.7: Regression coefficient estimates (in absolute value), and standard error (SE) of the estimates obtained after using MI-GSPCR, MI-Standard, and CC.

the number of iterations for MI-Standard still required less imputation time than using the current implementation of MI-GSPCR.

This highlights the biggest limitation of MI-GSPCR, namely the imputation time. MI-GSPCR uses intensive matrix manipulations at every iteration of the MICE algorithm and is therefore necessarily more computationally demanding. Furthermore, the cross-validation procedure used to define the values of the tuning parameters involves nested for-loops, which are known as computational bottlenecks, especially when written in the R language.

A significant improvement in the imputation time of MI-GSPCR could be achieved by dedicating more effort to optimizing the GSPCR algorithm and its software implementation. The *gpscr* package was written in R as a prototype for demonstrative purposes. Many implementation details could be changed to improve the efficiency of the R code (e.g., Gillespie & Lovelace, 2016), but even greater gains could be achieved by converting the code to a lower-level programming language, for example, C++. The multiple function calls and for-loops that are part of the current implementation of the GSPCR algorithm could benefit considerably from this change. Eddelbuettel and Balamuta (2018) showed how rewriting a function for bootstrap estimation of the mean and standard deviation of a distribution, involving a single for-loop, in C++ can halve the computation time. Eddelbuettel (2013, pp. 15–18) demonstrated an even more extreme example where rewriting the R code used to sample data for an auto-regressive process with C++ resulted in code that ran 130 times faster.

Apart from the computational limitations, the reader should also keep in mind that the performance of MI-GSPCR relies on other key implementation decisions that should be investigated further. For example, we defined the complexity of the model for the BIC to be equal to the number of parameters estimated by the GLM model regressing the observed values onto the principal components. One could also consider including the number of variables used to compute the components as part of this complexity. Further research is needed to investigate whether this decision has an impact on the imputation performance of MI-GSPCR.

Finally, while we have proposed MI-GSPCR as an alternative to traditional imputation model-building strategies, this approach does not need to be the only decision-making tool to select the predictors. When performing imputation to address missing values in data that will be used by researchers to estimate only a few known analysis models, it is desirable to use the analysis model as a main criterion to decide which predictors should be included in the imputation models. MI-GSPCR can be extended to directly include the analysis model predictors in their original form in the imputation models along with other predictors included through GSPCR.

5.5 Conclusions

By performing dimensionality reduction on a set of important predictors for every incomplete variable, MI-GSPCR allows the researcher to provide all variables in the data as possible predictors for the imputation models, which reduces the decision-making burden of specifying the MICE procedure. Despite this simplification of the overall complexity of using MICE, MI-GSPCR was able to deliver results on par with those obtained by more traditional specifications of MICE.

Chapter 6

Epilogue

This dissertation explored data-driven tools to facilitate the use of multivariate imputation by chained equations (MICE) for researchers analyzing survey data. We focused on the problem of selecting the predictors for the imputation models, which is one of the most delicate and important steps in specifying MICE. When using MICE to impute data, the analyst needs to specify an imputation model for every variable they want to impute, which entails selecting variables to use as predictors in the imputation model. A long-standing recommendation in the multiple imputation (MI) literature has been to include as many relevant predictors as possible in the imputation models. This inclusive strategy makes the missing at random assumption more plausible and can reduce the size of the standard errors in the analysis model (Collins et al., 2001). However, in practice, using complex imputation models with many predictors can lead to bias, poor efficiency, as well as estimation and convergence problems (e.g., Hardt et al., 2012; White et al., 2011). In this dissertation, we explored the use of prediction models that allow for a large number of independent variables as imputation models to maximize the advantages of the inclusive strategy while mitigating its limitations. We evaluated the quality of imputations obtained with these approaches based on the estimation bias and confidence interval coverage of a variety of parameters estimated from the imputed data. We focused on regression coefficients, correlations, means and variances as these parameters are part of common analyses performed in the social sciences. In the following discussion, we refer to the performance measures as bias and coverage for simplicity.

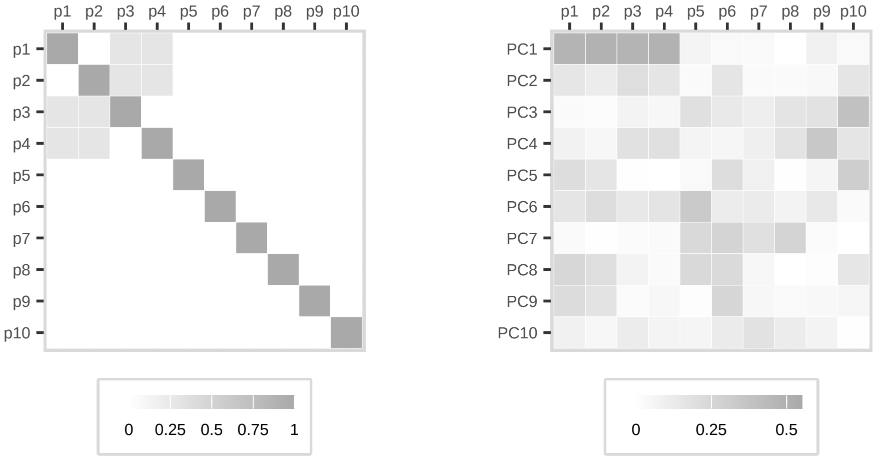
We proposed an approach to MI that uses a generalized version of supervised principal component regression as a univariate imputation model in MICE (MI-GSPCR). Principal component regression (PCR) is a dimensionality reduction technique that uses the principal components (PCs) of the independent variables instead of the independent variables in a regression model. Using PCR as an imputation model in MICE facilitates the choice of the imputation model predictors by allowing the analyst to provide all variables in the data as predictors in the imputation model without running into multicollinearity problems. Supervision makes sure that the PCs are relevant for the prediction of each variable under imputation. Generalization allows the imputation of both continuous and categorical data, based on both continuous and categorical predictors. When applied to real survey data, MI-GSPCR resulted in point estimates and standard errors similar to those obtained by more traditional approaches to variable selection, which can be considered a successful result. The goal of this dissertation was to work on an approach to MI that would achieve a similar quality of imputations to traditional ways of specifying the imputation algorithm, with a lower burden on the user, especially when analyzing data with hundreds of variables. This chapter provides an overview of the key contributions and limitations of the dissertation.

6.1 Number of PCs and supervision

The idea of using PCs obtained from principal component analysis (PCA) as predictors in the imputation models of MICE—which we refer to as MICE-PCA—has been explored in all the chapters of this dissertation. Chapters 2, 3 and 4 showed substantial differences in the bias and coverage properties of MICE-PCA depending on the number of PCs used. The significance of supervised PCR in MICE-PCA is that it provides a data-driven way of computing PCs that summarize the potential predictors and are relevant for the prediction of the variables under imputation. When used as an imputation model, supervised PCR makes sure that if MAR predictors are present in the data, the PCs that summarize them will be used as predictors in the imputation models. The results of Chapter 4 confirmed that all types of supervised dimensionality reduction we considered improved upon the use of standard PCA to generate good predictors for the imputation models in MICE. Re-examining the results from Chapters 2 and 3 corroborates this finding.

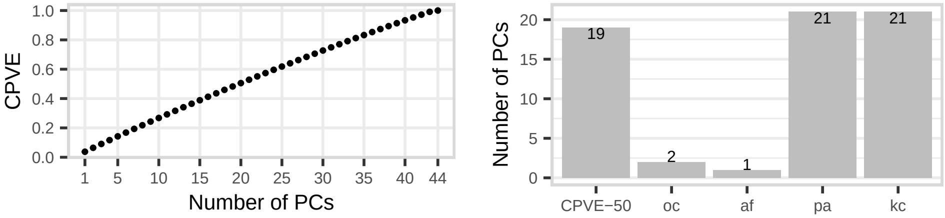
In the simulation study presented in Chapter 2, in the low-dimensional data condition, we generated six variables with missing values and 44 potential predictors for the imputation models. Among the potential predictors, only four variables were important for the imputation procedure, which we referred to as the missing at random predictors (MAR predictors). The remaining 40 variables were unimportant for the imputation task, meaning that their inclusion as predictors in the imputation models provided no benefit. We used PCA to generate PCs based on all of the 44 potential predictors, kept the number of components that explained 50% of their variance, and found that MICE-PCA resulted in low bias and good coverage.

In response to feedback we received during the peer review process, we carried out additional analyses to consider what would happen for different degrees of collinearity in the data. When we increased the correlation among the 40 unim-



(a) Correlation matrix of the first 10 potential predictors (four MAR predictors, p_1 to p_4, and six unimportant predictors).

(b) Loadings of the first 10 potential predictors on the first 10 PCs.



(c) Cumulative proportion of explained variance by the 44 PCs in the data.

(d) Number of PCs that would be selected using the 50% cumulative explained variance rule (CPVE-50), the optimal coordinates index (*oc*), the acceleration factor (*af*), the parallel analysis criterion (*pa*), and the Kaiser criterion (*kc*).

Figure 6.1: PCA results for the potential predictors in the simulation study in Chapter 2 for the condition with no collinearity among unimportant predictors.

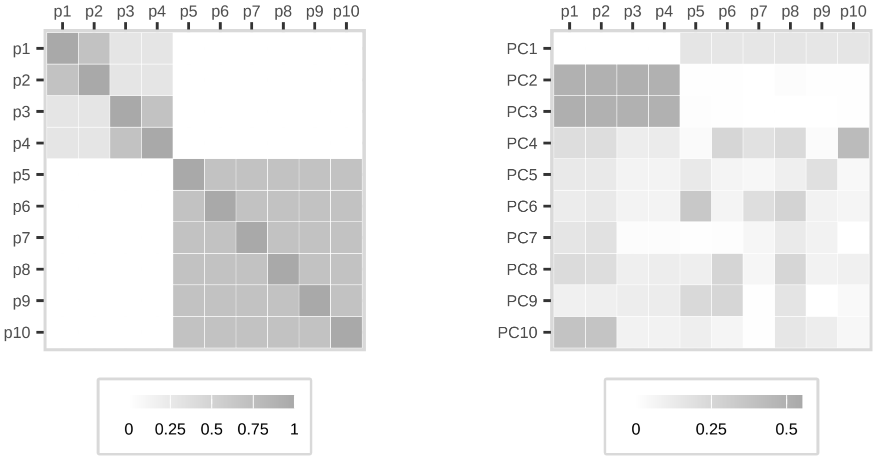
portant predictors—as described in Section 2.3.3—we discovered that MICE-PCA resulted in larger bias and worse coverage. To understand this result, it is important to consider how the increased correlation between the unimportant predictors influenced PCA.

In Figure 6.1, we report the correlation matrix of the first 10 potential predictors (i.e., the four MAR predictors—p1, p2, p3, and p4—together with six unimportant predictors) in the original simulation study, where the 40 unimportant predictors were uncorrelated. In the same figure, we report the absolute value of the loadings of the first 10 potential predictors on the first 10 PCs, the cumulative proportion of variance explained by the 44 PCs in the data, and the number of PCs that are selected by five alternative non-graphical decision rules—the 50% cumulative explained variance rule (CPVE-50) that we used in the simulation study, the optimal coordinates index (*oc*), the acceleration factor (*af*), the parallel analysis criterion (*pa*), the Kaiser criterion (*kc*). In the original study, 19 PCs were needed to summarize 50% of the variance in the set of 44 potential predictors. Among these PCs, the first one had high loadings for the four MAR predictors, effectively summarizing the relevant information for imputation.

In Figure 6.2, we report the same information for a condition where the correlation between the 40 unimportant predictors was high (0.7). The first PC summarized 60% of the variance in the data, but the MAR predictors did not meaningfully contribute to the component score (loadings close to 0). In this condition, the leading PC mostly captured the variance in the large block of unimportant but correlated predictors and the MAR predictors only loaded on the second and third PCs. By selecting the PCs that explained 50% of the variance in the potential predictors, MICE-PCA used only the leading PC in the imputation models, effectively excluding the MAR predictors, and therefore it resulted in larger bias and worse coverage.

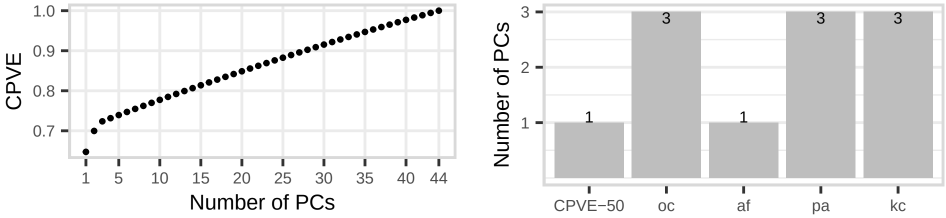
The results from Chapter 2 suggested that the number of PCs used by MICE-PCA should be selected carefully. The 50% rule proved to be unsatisfactory and we recommended exploring alternative decision rules. From Figures 6.1d and 6.2d it is clear that had we used a different decision rule, we would have selected the relevant PCs summarizing the MAR predictors even in the conditions with a stronger correlation between the unimportant predictors. The oc, pa and kc rules would have selected the first three PCs in the high correlation condition, which would have included the PCs summarizing variation on the MAR predictors in the imputation models. However, this is only an ad hoc solution. It is easy to imagine a data set where one or more MAR predictors are not part of any of the major blocks of variance in the data. In such a case, any of the alternative decision rules considered would still be unsatisfactory.

The importance of defining a good decision rule for the number of PCs to use in MICE-PCA becomes clearer when re-examining the results of Chapter 3, where we explored directly the relationship between the quality of imputations obtained with MICE-PCA and the number of PCs used. In the simulation study, seven or more PCs were needed as predictors for the imputation models to obtain good imputations. This result was interesting because we generated data based on exactly seven latent factors, which suggested that to use MICE-PCA effectively we needed



(a) Correlation matrix of the first 10 potential predictors (four MAR predictors, p_1 to p_4, and six unimportant predictors).

(b) Loadings of the first 10 potential predictors on the first 10 PCs.



(c) Cumulative proportion of explained variance by the 44 PCs in the data.

(d) Number of PCs that would be selected using the 50% cumulative explained variance rule (CPVE-50), the optimal coordinates index (*oc*), the acceleration factor (*af*), the parallel analysis criterion (*pa*), and the Kaiser criterion (*kc*).

Figure 6.2: PCA results for the potential predictors in the simulation study in Chapter 2 for a condition with high collinearity among unimportant predictors.

to use enough PCs to capture the latent structure of the data. However, based on the re-examination of the collinearity results from Chapter 2, the situation seems to be more nuanced.

In the first simulation study in Chapter 3, we generated data based on a factor analysis model with seven latent factors. The first latent factor was measured by eight variables, four of which were MAR predictors for the missingness of the other four. The remaining six latent factors were measured by eight variables each. As a result, the variables measuring the first latent factor constituted a block of correlated variables, while the six groups of eight variables measuring the other latent factors constituted six other blocks of correlated variables. When imputing any of the variables with missing values in these data, seven variables from the first block and eight from each of the other six blocks were available as potential predictors for the imputation model. When performing PCA on these potential predictors, because all variables are standardized, the first six PCs summarize variance on the six blocks of eight variables, while the seventh PC summarizes variance from the block of seven variables. Therefore, it is possible that seven or more PCs were needed to obtain good imputations because only the seventh PC captured information about the MAR predictors, rather than because enough PCs were needed to capture the latent structure of the data. Repeating the simulation studies in Chapter 3 with an additional condition where only the seventh PC is used as a predictor in the imputation model could show whether this interpretation of the results is correct.

The reflection on the results of Chapters 2 and 3 gives a clear picture of what can go wrong when using PCs as predictors in the imputation models. The retained PCs might summarize almost all variance in the set of potential predictors, but if the MAR predictors do not substantially contribute to any of these PCs, then the PCs do not capture the important variation for the imputation task. The analyst must make sure that the PCs used as predictors in the imputation models capture the MAR predictors. However, none of the decision rules for the number of PCs considered in Chapters 2 and 3 are satisfactory because they focus on finding the number of PCs that best summarize the potential predictors, disregarding whether they capture the information that is important for the imputation task.

6.2 MI-GSPCR in practice

In Chapter 5, we adapted the use of supervised PCR to allow the imputation of data consisting of a mix of continuous and categorical data, a fundamental characteristic of survey data. We expanded upon the supervised PCR approach proposed by Bair et al. (2006) by extending its logic for data consisting of binary, ordinal, and nominal variables and we implemented an algorithm to estimate this generalization of supervised PCR in an R package (Costantini, 2023b). We also created experimental functions in the *mice* R package to demonstrate the use of GSPCR as an imputation method for MI. The resulting MI-GSPCR approach is easy to use to impute survey data with hundreds of variables, as we showed in the tutorial in Appendix B, and it results in similar imputations to those obtained by carefully selecting the predictors for the imputation models. Alongside these strengths, social scientists who would like to use MI-GSPCR should keep in mind the following limitations.

Chapter 5 highlighted the computational intensity of the current implementation of MI-GSPCR. As we discussed in Section 5.4, the current implementation of GSPCR was written in R, which favored easily interpretable code over performance. Apart from re-writing the code base in a more efficient programming language, future research could consider reducing the frequency of updating the principal components. Part of the computational intensity of MI-GSPCR is rooted in the repeated estimation of PCA at every iteration, for every variable under imputation. As discussed in in Chapter 3, the decision to re-estimate PCs at every iteration came out of the desire to estimate PCA on originally incomplete potential predictors, without requiring preprocessing decisions on how to handle missing values on the potential predictors of the imputation models. However, whether it is necessary to update these PCs at every single iteration of the MICE algorithm remains to be seen. We anticipate that with modest proportions of missing data per variable (e.g., 0.1 or less), the difference between two PCs computed in two subsequent iterations might be negligible. Future research could explore the conditions under which it is acceptable to reduce the frequency of estimating PCA in MI-GSPCR.

We mostly worked with a single block of cross-sectional data. However, social scientists are often faced with linked data (i.e., blocks of variables coming from multiple data sources), longitudinal data, or a combination of both. For example, a researcher might have at their disposal questionnaire data, experience sampling data, and genetic data on the same subjects. Linking these different data blocks together can discover new connections between social, behavioral, and biological factors, but poses two challenges for MI-GSPCR. First, the large number of variables that are collected from multiple sources can result in high-dimensional settings (i.e., more variables than observations), and parameter estimates for PCA are statistically consistent only if the data set is low-dimensional (Shen et al., 2016), a problem we have not considered in this dissertation. Second, common mechanisms between blocks of data can be lost when ignoring the multi-source nature of the data because mechanisms that are distinctive of each data source are often more pronounced (de Schipper & Van Deun, 2019; Van Deun et al., 2009). These issues can be taken into account to create predictors of higher quality for the imputation models when MI-GSPCR is applied to linked data. Future research could consider extensions of PCA that help identify sparse common and distinctive components between blocks of data to define the predictors of the imputation models for multi-block data, especially in their supervised form (S. Park et al., 2021, 2023).

Finally, social scientists often analyze nested data, where different observations are clustered within groups. For example, when respondents to a survey are grouped within different countries, respondents from the same country may be more similar to one another than to respondents from other countries. Similarly, in longitudinal data, the same respondent is asked the same questions multiple times which results in clustering of observations within respondents. In all of the example analyses of the EVS data, we treated the clustering of observations within countries by coding this grouping variable as a series of dummy variables and extracting PCs from the larger data matrix. Andridge (2011), Enders et al. (2016), Lüdtke et al. (2017), and van Buuren (2011) have shown the limits of using dummy variables to account for clustered observations for multilevel imputation. Future research should consider how MI-GSPCR and the use of PCA interact with clustered observations more carefully. Perhaps the use of multilevel principal component analysis (Timmerman, 2006) could prove useful in extending the MI-GSPCR for imputation of multilevel data.

Appendices

A The GSPCR approach

In the GSPCR approach, a dependent variable is regressed onto several PCs that are estimated from a subset of relevant predictors in the data. Consider a data matrix $\mathbf{X}$ with N rows and P columns, where the rows represent different response units and the columns represent variables, which can have different measurement levels. Consider a dependent variable $\mathbf{y}$ of any measurement level.

1. Regress $\mathbf{y}$ onto each variable in $\mathbf{X}$ via P separate GLMs with an appropriate random component and link function
2. Compute the fit measures for each of the P GLMs.
3. For each value of a threshold ρ_l from the list $\rho_1 < \rho_2 < \cdots < \rho_L$
   a) Define the subset $\mathbf{X}_s \in \mathbf{X}$ by discarding all variables whose GLM fit measure is smaller than ρ_l (or bigger than ρ_l, depending on the chosen fit measure).
   b) Compute the first Q PCs of $\mathbf{X}_s$
   c) Regress the $\mathbf{y}$ variable on these PCs
4. Pick the value of ρ_l and Q by cross-validation.

The approach is supervised because it uses the relationship between the dependent variable and the other variables in the data to select the best variables for the PCR model. The choice of parameters ρ and Q is made by a K-fold cross-validation.

GSPCR is based on SPCR but they differ in three ways. Firstly, SPCR defines ρ as the regression coefficient in the simple linear regression of the dependent variable and each predictor to identify the active set of predictors to use in computing the PCs. However, this limits the possible predictors to numeric variables. Categorical predictors are included in regression models through dummy variables or alternative coding schemes, meaning that simple linear regression with categorical predictors with more than two levels results in two or more regression coefficients, making it difficult to compare the strength of the association between numeric predictors and categorical predictors. GSPCR uses the Cox and Snell R-squared (R^2_{CS}; Nagelkerke, 1991), which is a scaled ratio between the likelihood under the intercept-only model (L_0) and the estimated model (L_M),

$$R^2_{CS} = 1 - \left(\frac{L_0}{L_M} \right)^{2/n},$$

where n is the sample size of the data. Using R^2_{CS} as a measure of effect size allows the comparison of the strength of numeric and categorical predictors. Compared to the traditional multiple correlation coefficient R^2, R^2_{CS} can be computed for continuous, binary, ordinal, or nominal dependent variables, but it cannot always be interpreted as a proportion of variance explained, and its maximum value is $L_0^{2/n}$, which can be smaller than 1.

The second difference between GSPCR and SPCR is that to allow categorical predictors in $\mathbf{X}$, GSPCR uses PCAmix (Chavent et al., 2012, 2014) to compute the PCs in step 3b. PCAmix combines PCA and MCA to weigh continuous and categorical predictors so that neither type will dominate the other in the computation of the PCs because of scaling differences. The third difference between SPCR and GSPCR is the type of fit measure used for the cross-validation procedure. SPCR uses the F-statistic as a fit measure for the cross-validation procedure for linear regression and the likelihood ratio test for the proportional hazards model for survival data (Cox, 1972; Prentice & Gloeckler, 1978). As a consequence, SPCR does not support other model forms and variable measurement levels. GSPCR can use the Bayesian information criterion (BIC; Smith & Spiegelhalter, 1980), the Akaike information criterion (AIC; Akaike, 1998), and R^2_{CS} as fit measures for the cross-validation procedure. Any of these fit measures are applicable for all types of dependent variables, but BIC and AIC also account for model complexity, which allows choosing the values of ρ_l and Q without K-fold cross-validation.

B How to use MI-GSPCR in R

In this tutorial, we show how to generate multiple imputations with generalized supervised principal component regression (MI-GSPCR) using an experimental version of the R package *mice* (van Buuren & Groothuis-Oudshoorn, 2011). Note that this experimental version was developed by Edoardo Costantini as a fork of the original repository. In case of any issues replicating the analysis reported here, please contact the main author or open an issue on the GitHub repository hosting the code for the forked project[23].

For this tutorial, we used a synthetic data set generated based on 33 variables from the European Values Study (EVS, EVS, 2020a). EVS data are a mix of binary, ordinal, and nominal variables, which is common for social scientists. The data were synthesized (Raghunathan, 2021; Volker & Vink, 2021) to facilitate distribution and are available in the permanent repository stored at the DOI provided in Costantini (2024a).

B.1 Installation and setup

The experimental version of *mice* used in this tutorial can be installed from its GitHub repository using the *remotes* package. The user should be aware that this will replace any version of *mice* installed on their computer and they might prefer to install this version in a different local library instead.

```r
# Install package from the GitHub repository
remotes::install_github(
    "https://github.com/EdoardoCostantini/mice/tree/develop-gspcr"
)
```

If the link does not work anymore, the software can always be retrieved by downloading `mice_3.16.0.0012.tar.gz` from the DOI provided in Costantini (2024b).

```r
# Optional: install from local folder
install.packages(
  "/path/to/package.tar.gz",
  repos = NULL,
  type = "source"
)
```

Once installed, load the experimental version of *mice* as any other package:

---

[23] https://github.com/EdoardoCostantini/mice/issues

```
# Load the experimental version of mice
library(mice)
```

The following warning message will appear in the console:

```
#> mice 3.16.0.0012 2023-11-28 /opt/homebrew/lib/R/4.4/site-library was loaded.
#>
#> This is an experimental version of the `mice` R package
#> developed by Edoardo Costantini as an unofficial fork to the
#> main package. You can find the code base by following this link:
#>
#> https://github.com/EdoardoCostantini/mice/tree/develop-gspcr
#>
#> I created this version to demonstrate the potential use of
#> generalised supervised principal components as a univariate
#> imputation method in `mice`.
#>
#> IMPORTANT!
#>
#> I created this version in autonomy from maintainers and
#> other contributors of the CRAN version of the `mice` R package.
#> If you encounter any issues using it please open an issue on
#> the GitHub repository hosting the code for the forked project:
#>
#> https://github.com/EdoardoCostantini/mice/issues
```

which reminds the user this version is an unofficial experiment developed only for demonstrative purposes and independently from the CRAN version of *mice*.

Next, install the *gspcr* (Costantini, 2023b) R package, the engine behind MI-GSPCR. The package is available on CRAN:

```
# Install gspcr
install.packages("gspcr")
```

Alternatively, to install the latest developmental version, use:

```
# Install development version of gspcr from the GitHub repository
remotes::install_github(
    "https://github.com/EdoardoCostantini/gspcr"
)
```

Load the package:

```
# Load gspcr
library(gspcr)
```

Finally, install and load the *ggmice* and *ggplot2* packages which provide useful functions to generate imputation diagnostic plots.

```r
# Load other packages we will need
library(ggmice)
library(ggplot2)
```

B.2 Imputation methods

The GSPCR imputation model is implemented in *mice* through five additional univariate imputation methods (see `?mice` for a list of all the supported univariate imputation methods):

- `mice.impute.gspcr.pmm` is meant to impute ordinal and continuous variables with non-normal distributions (e.g., skewed, multi-modal) and was inspired by the default *mice* imputation method `mice.impute.pmm`.
- `mice.impute.gspcr.norm` is meant to impute continuous normally distributed variables (based on `mice.impute.norm.boot`).
- `mice.impute.gspcr.logreg` is meant to impute binary variables (based on `mice.impute.logreg`).
- `mice.impute.gspcr.polr` is meant to impute ordered categorical variables (based on `mice.impute.polr`).
- `mice.impute.gspcr.polyreg` is meant to impute *unordered* categorical variables (based on `mice.impute.polyreg`).

Each of these imputation methods uses GSPCR as the modelling tool to obtain imputations. Uncertainty regarding the parameters of the imputation model is incorporated by taking a bootstrap sample of the data before estimating GSPCR, a standard way of obtaining proper multiple imputations (van Buuren, 2018, pp. 67-69). The additional functions take five GSPCR-related arguments. The default values for these arguments are applicable for the imputation of data with variables of various measurement levels, so the user does not need to change them. However, here we describe these GSPCR arguments in more detail for the interested reader:

1. `thrs`—GSPCR computes a measure of bivariate association between every variable to be imputed and every variable in the data. These associations are used to define a set of important predictors that should be used when computing the principal components (PCs) that will be used as predictors for the variables under imputation. The default value for this argument is `PR2`, which

is the Cox and Snell Pseudo R-squared described in Appendix A. However, the current version of the *gspcr* package used supports even more association measures (see `help(cv_gspcr)`, the help file for the function used to estimate the GSPCR models, for more details).

2. `nthrs`—For a given association type, the smallest and highest association between the variable under imputation and all the variables are computed. This range is then divided into equally spaced steps to define a vector of association values to be considered by the model parameter tuning procedure. The number of steps is defined by the integer `nthrs`.
3. `npcs_range`—GSPCR uses the active set of predictors based on a certain association threshold to compute Q PCs. The range of PCs to be used is defined by the argument `npcs_range`. By default, only the first three components are considered as the supervised nature of GSPCR favors a small number of PCs. Because this argument specifies only the range of PCs to be used in the tuning procedure, the number of PCs selected might be smaller than the highest value provided in `npcs_range`.
4. `fit_measure`—The criterion used for the tuning procedure. The default value is set to `BIC`, a criterion that describes the goodness of fit for the GSPCR models with different numbers of PCs and threshold values and accounts for the model complexity. The current version of the *gspcr* package also supports additional criteria (see `help(cv_gspcr)`).
5. `K`—The number of folds for the K fold cross-validation procedure. The default value is set to `K = 1`, meaning that the fit measures are computed on the training data. The BIC criterion already accounts for model complexity which allows us to save time by avoiding the division in folds for the cross-validation procedure.

The help files provide a more detailed description of the arguments and the values they can take.

```r
# Check out the help file for the methods
help("mice.impute.gspcr.pmm")
help("mice.impute.gspcr.norm")
help("mice.impute.gspcr.logreg")
help("mice.impute.gspcr.polr")
help("mice.impute.gspcr.polyreg")
```

B.3 Example MI-GSPCR with EVS data

Once all required packages have been installed and loaded, the user can carry out the example run of the MI-GSPCR using synthetic EVS data. Download the synthetic EVS data from the DOI provided in Costantini (2024a) and load them by using the following command:

```
# Load the data
EVS <- readRDS("/path/to/EVS.rds")
```

Explore the percentage of missing values per variable:

```
# Percentage of missing values per variable
round(colMeans(is.na(EVS)), 3)*100

#>      v174_LR         v225     v246_egp         v185         v186         v187
#>          9.0          0.0          0.0          2.7          2.6          4.0
#>         v145         v110          v97          v98          v99         v100
#>          6.7          1.3          0.0          0.9          2.2          1.0
#>         v101       age_r3 v243_ISCED_1         v234          v54        v52_r
#>          1.6          0.0          0.6          0.2          0.2          0.4
#>           v1           v2           v3           v4           v5           v9
#>          0.8          0.4          0.1          0.2          0.5          0.5
#>          v10          v11          v12          v13         v111         v113
#>          0.8          0.5          0.6          0.7          2.4          5.9
#>         v114         v204         v267
#>          3.7          6.7          0.0
```

All variables have missing values except for v225, v246_egp, v97, age_r3, and v267.

B.4 Type of imputation models

To perform any imputation with *mice*, the appropriate type of imputation model (e.g., linear or logistic regression) need to be defined for each variable under imputation. To do so, check what types of variables are in the data by exploring the classes of each variable stored in `EVS`:

```
# Check out variable classes
classes <- sapply(EVS, function(x) class(x)[1])

# Check the distribution of classes
table(classes)

#> classes
#>  factor ordered
#>      21      12
```

There are 21 unordered factors and 12 ordered factors. To differentiate between binary and multi-categorical variables, count the number of levels in each factor:

```
# Count the number of levels
categories <- sapply(EVS, nlevels)

# Print the variable levels counts
table(categories, classes)

#>           classes
#> categories factor ordered
#>         2      6       0
#>         3     10       4
#>         4      4       3
#>         7      1       2
#>         10     0       3
```

This data exploration clarifies that there are 6 binary factors, 12 ordered factors, and 15 unordered factors in the data. Treat factors with two levels as binary variables, ordered factors as ordinal variables, and unordered factors with more than two categories as nominal variables by using `gspcr.logreg`, `gspcr.pmm`, and `gspcr.polyreg`, respectively, as imputation methods.

To define these imputation methods, create a method vector to pass as an argument to the `mice::mice()` function. Start by creating a vector of default methods using the `mice::make.method()` function.

```
# Create empty method vector
meths <- mice::make.method(EVS)
```

Then, assign the type of imputation model based on the class type and the number of categories and remove the imputation model for the variables that are fully observed.

```
# Define methods
meths[classes == "ordered"] <- "gspcr.pmm"
meths[classes == "factor" & categories == 2] <- "gspcr.logreg"
meths[classes == "factor" & categories > 2] <- "gspcr.polyreg"

# Remove methods for fully observed variables
meths[colSums(is.na(EVS)) == 0] <- ""
```

Check that the desired methods have been specified correctly by printing the variable names with their class, number of categories and corresponding imputation method:

```
# Check methods
data.frame(classes, categories, meths)

#>              classes categories         meths
#> v174_LR      ordered          7     gspcr.pmm
#> v225          factor          2
#> v246_egp      factor          7
#> v185         ordered         10      gspcr.pmm
#> v186         ordered         10      gspcr.pmm
#> v187         ordered         10      gspcr.pmm
#> v145         ordered          4      gspcr.pmm
#> v110          factor          4  gspcr.polyreg
#> v97          ordered          4      gspcr.pmm
#> v98           factor          3  gspcr.polyreg
#> v99           factor          3  gspcr.polyreg
#> v100          factor          3  gspcr.polyreg
#> v101          factor          3  gspcr.polyreg
#> age_r3       ordered          7
#> v243_ISCED_1  factor          3  gspcr.polyreg
#> v234          factor          3  gspcr.polyreg
#> v54           factor          3  gspcr.polyreg
#> v52_r         factor          4  gspcr.polyreg
#> v1           ordered          3      gspcr.pmm
#> v2           ordered          3      gspcr.pmm
#> v3           ordered          3      gspcr.pmm
#> v4           ordered          3      gspcr.pmm
#> v5           ordered          4      gspcr.pmm
#> v9            factor          2   gspcr.logreg
#> v10           factor          2   gspcr.logreg
#> v11           factor          2   gspcr.logreg
#> v12           factor          2   gspcr.logreg
#> v13           factor          2   gspcr.logreg
#> v111          factor          4  gspcr.polyreg
#> v113          factor          3  gspcr.polyreg
#> v114          factor          3  gspcr.polyreg
#> v204          factor          3  gspcr.polyreg
#> v267          factor          4
```

B.5 Perform imputation

To impute with MI-GSPCR, call the `mice::mice()` function by providing the `meths` vector for the `method` argument. No matter the number of columns in the original data, there is no need to specify which predictors to use in the 33 imputation models because all possible predictors in the EVS data are handled by the GSPCR imputation models. However, the `predictorMatrix` argument in the `mice::mice()` call can still be used to force any variable out of the pool of predictors (see `?make.predictorMatrix()`).

```
# Sequential MICE run
mids_migspcr <- mice::mice(
    # General mice arguments
    data = EVS,
    m = 5,
    maxit = 10,
    method = meths,
    seed = 20230804,
    # Take control of some internal processes
    use.matcher = TRUE, # use matcher for replicability
    ridge = 0, # no ridge penalty
    eps = 0, # bypasses remove.lindep()
    threshold = 1L, # no collinearity checks needed
    # GSPCR specific arguments
    thrs = "PR2",
    fit_measure = "BIC",
    nthrs = 5,
    npcs_range = 1:3,
    K = 1
)
```

In this call of the `mice::mice()` function, the first five arguments are standard: the data, the number of multiple imputations (`m = 5`), the number of iterations (`maxit = 10`), the methods vector (`method = meths`), and the (pseudo) random number seed (`seed = 20230804`) for replicability purposes. The GSPCR arguments described above are declared for clarity but the user may leave them set to their default values. Finally, the arguments `ridge`, `eps`, and `threshold` were specified to avoid any collinearity check performed by the `mice::mice()` engine as these measures are not needed when using *gspcr*-based imputation methods.

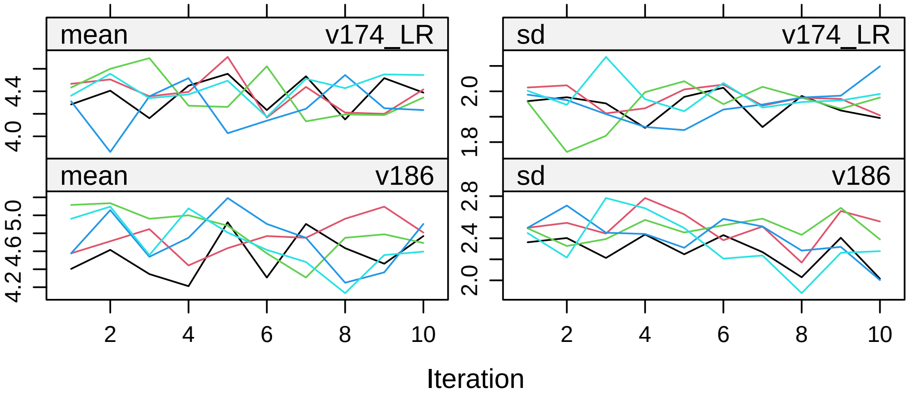
Even for this simple example, this call will take a long time to run. Hence, it is recommended to use the `mice::futuremice()` version of `mice::mice()` to parallelize the imputation procedure.

```
# Parellel MICE run
mids_migspcr <- futuremice(
    # Parallel specific arguments
    parallelseed = 20240410,
    n.core = 5,
    # General mice arguments
    data = EVS,
    m = 5,
    maxit = 10,
    method = meths,
    use.matcher = TRUE, # use matcher for replicability
    ridge = 0,
    eps = 0, # bypasses remove.lindep()
    threshold = 1L,
    # GSPCR specific arguments
    thrs = "PR2",
    fit_measure = "BIC",
    nthrs = 5,
    npcs_range = 1:3,
    K = 1
  )
```

B.6 Diagnostic measures

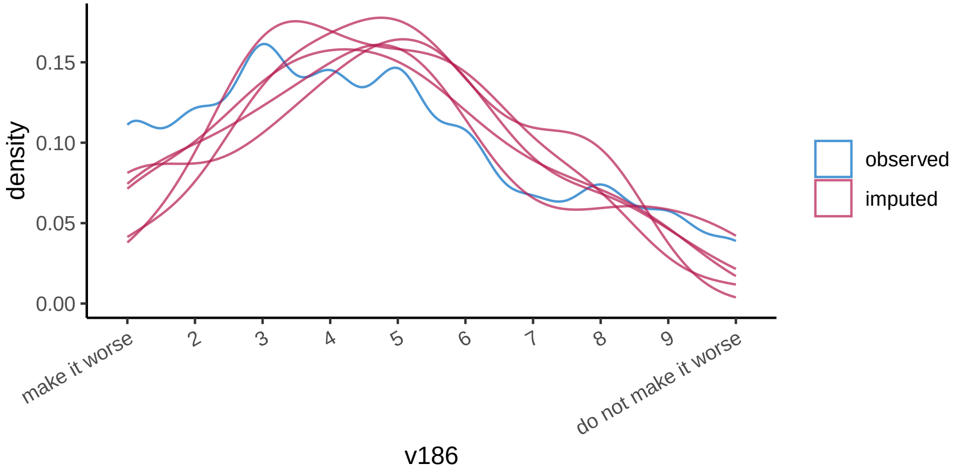
After imputation, the object `mids_migspcr` is created. This object is of the `mids` class, so all of the functions and workflow from the `mice::mice()` can be used to perform diagnostic checks and estimation of analysis models. First, check the convergence of the algorithm by looking at the trace plots for all the imputed variables.

```
# Exemplifying convergence checks for two imputed variables
plot(mids_migspcr, c("v174_LR", "v186"))
```



The *ggmice* framework can also be used to plot diagnostic measures. For example, we can compare the distribution of imputations and observed values with `geom_density()`:

```
# Density plots
ggmice::ggmice(
    data = mids_migspcr,
    ggplot2::aes(x = v186, group = .imp)) +
    ggplot2::geom_density() +
    ggplot2::theme(
        legend.position = "right",
        axis.text.x = ggplot2::element_text(
            angle = 30,
            vjust = 1,
            hjust = 1
        )
    )
```



B.7 Substantive analysis of the imputed data

The analysis model can be estimated on the imputed data in the same way as with any other `mids` object. For this example, start by collecting the imputation in a single long `data.frame`.

```
# Extract the data set with imputed values in long format
long_migspcr <- mice::complete(mids_migspcr, "long", include = TRUE)
```

This format makes it easy to process variables in desired ways. For example, the ordinal left-right voting tendency can be transformed to a numeric variable, and the *Political action* scale can be created as the sum score obtained by adding variables v98 to v101.

```
# Transform variables to versions you want to use for analysis
long_list_processed <-
    within(
        data = long_migspcr,
        expr = {
            LF_voting <- as.numeric(v174_LR)
            Gender <- v225
            Political_interest <- as.numeric(v97)
            Political_action <- (
                as.numeric(v98) +
                as.numeric(v99) +
                as.numeric(v100) +
                as.numeric(v101)
            ) / 4
        }
    )
```

After processing the data, reshape the imputations to a `mice::mids` object:

```
# Revert to a mids object
mids_processed <- mice::as.mids(long_list_processed)
```

and finally fit some analysis model of interest.

```
# Fit an analysis model
fits <- with(
  mids_processed,
  lm(LF_voting ~ Gender + Political_interest + Political_action)
)
```

The estimates of the parameters in the analysis model can be pooled with the standard `mice::pool()` function.

```
# Pool estimates between different model fits
fits_pooled <- mice::pool(fits)

# Print the results
summary(fits_pooled)

#>                 term    estimate  std.error  statistic       df      p.value
#> 1        (Intercept)  2.26734410 0.30002204  7.5572584 726.3857 1.242347e-13
#> 2       Genderfemale  0.08266419 0.08874427  0.9314877 835.4711 3.518704e-01
#> 3 Political_interest -0.01740460 0.05341617 -0.3258302 552.8379 7.446761e-01
#> 4   Political_action  0.92007060 0.10838185  8.4891573 228.9506 2.636578e-15
```

Running the code in this tutorial may result in slightly varied imputations, leading to different pooled parameter estimates. These differences should not impact interpretation. The *gspcr* package uses the built-in R function `base::svd()` to compute the singular value decomposition used to estimate the PCs in the imputation models. At the time of writing, this function has some known convergence issues[24,25,26] and it can lead to different solutions based on the computer's architecture in certain edge cases. Measures to mitigate numerical instability will be the focus of the next developmental phases of *gspcr*.

---

[24] https://github.com/Reference-LAPACK/lapack/issues/672#issue-1241498134

[25] https://github.com/husson/FactoMineR/blob/2e5414d8c1724aaab655a2ed914235b0227ee504/R/svd.triplet.R#L23

[26] https://github.com/EdoardoCostantini/gspcr/issues/27

References

Abraham, G., & Inouye, M. (2014). Fast principal component analysis of large-scale genome-wide data. *PloS One*, *9*(4), e93766. https://doi.org/10.1371/journal.pone.0093766

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike* (pp. 199–213). Springer. https://doi.org/10.1007/978-1-4612-1694-0_15

Andridge, R. R. (2011). Quantifying the impact of fixed effects modeling of clusters in multiple imputation for cluster randomized trials. *Biometrical Journal*, *53*(1), 57–74. https://doi.org/10.1002/bimj.201000140

Bair, E., Hastie, T., Paul, D., & Tibshirani, R. (2006). Prediction by supervised principal components. *Journal of the American Statistical Association*, *101*(473), 119–137. https://doi.org/10.1198/016214505000000628

Bartholomew, D. J., Knott, M., & Moustaki, I. (2011). *Latent variable models and factor analysis: A unified approach* (3rd ed.). John Wiley & Sons. https://doi.org/10.1002/9781119970583

Bollen, K. A. (1989). *Structural equations with latent variables*. John Wiley & Sons. https://doi.org/10.1002/9781118619179

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Brick, J. M., & Williams, D. (2013). Explaining rising nonresponse rates in cross-sectional surveys. *The ANNALS of the American Academy of Political and Social Science*, *645*(1), 36–59. https://doi.org/10.1177/0002716212456815

Burgette, L. F., & Reiter, J. P. (2010). Multiple imputation for missing data via sequential regression trees. *American Journal of Epidemiology*, *172*(9), 1070–1076. https://doi.org/10.1093/aje/kwq260

Burton, A., Altman, D. G., Royston, P., & Holder, R. L. (2006). The design of simulation studies in medical statistics. *Statistics in Medicine*, *25*(24), 4279–4292. https://doi.org/10.1002/sim.2673

Chavent, M., Kuentz-Simonet, V., Labenne, A., & Saracco, J. (2014). Multivariate analysis of mixed data: The R package PCAmixdata. *arXiv preprint*. https://doi.org/10.48550/arXiv.1411.4911

Chavent, M., Kuentz-Simonet, V., & Saracco, J. (2012). Orthogonal rotation in pcamix. *Advances in Data Analysis and Classification*, *6*(2), 131–146. https://doi.org/10.1007/s11634-012-0105-3

Chung, D., & Keles, S. (2010). Sparse partial least squares classification for high dimensional data. *Statistical Applications in Genetics and Molecular Biology*, *9*(1). https://doi.org/10.2202/1544-6115.1492

Collins, L. M., Schafer, J. L., & Kam, C.-M. (2001). A comparison of inclusive and restrictive strategies in modern missing data procedures. *Psychological Methods*, *6*(4), 330–351. https://doi.org/10.1037//1082-989X.6.4.330

Costa Jr, P. T., & McCrae, R. R. (2008). The revised NEO personality inventory (NEO-PI-R). In *The sage handbook of personality theory and assessment: Volume 2 - personality measurement and testing* (pp. 179–198, Vol. 2). Sage Publications, Inc. https://doi.org/10.4135/9781849200479.n9

Costa Jr, P. T., McCrae, R. R., & Dye, D. A. (1991). Facet scales for agreeableness and conscientiousness: A revision of the NEO personality inventory. *Personality and Individual Differences*, *12*(9), 887–898. https://doi.org/10.1016/0191-8869(91)90177-D

Costantini, E. (2023a). *Fireworks* (Version v2.1). Zenodo. https://doi.org/10.5281/zenodo.7529333

Costantini, E. (2023b). *Gspcr: Generalized supervised principal component regression* [R package version 0.9.4.1]. https://CRAN.R-project.org/package=gspcr

Costantini, E. (2023c). *Mi-hd* (Version v2.1). Zenodo. https://doi.org/10.5281/zenodo.8246041

Costantini, E. (2023d). *Mi-pca* (Version v2.1). Zenodo. https://doi.org/10.5281/zenodo.7529273

Costantini, E. (2023e). *Mi-spcr* (Version v3.0.1). Zenodo. https://doi.org/10.5281/zenodo.8379661

Costantini, E. (2023f). *Plotmigspcr*. Zenodo. https://doi.org/10.5281/zenodo.8383559

Costantini, E. (2023g). *Plotmispcr*. Zenodo. https://doi.org/10.5281/zenodo.7451801

Costantini, E. (2024a). *Evs-lookalike*. Zenodo. https://doi.org/10.5281/zenodo.10599480

Costantini, E. (2024b). *Mi-gspcr* (Version v1.2). Zenodo. https://doi.org/10.5281/zenodo.10599323

Costantini, E. (2024c). *Plotmihd*. Zenodo. https://doi.org/10.5281/zenodo.7928442

Costantini, E. (2024d). *Plotmipca*. Zenodo. https://doi.org/10.5281/zenodo.7391522

Costantini, E., Lang, K. M., Reeskens, T., & Sijtsma, K. (2023). High-dimensional imputation for the social sciences: A comparison of state-of-the-art methods. *Sociological Methods & Research*. https://doi.org/10.1177/00491241231200194

Costantini, E., Lang, K. M., & Sijtsma, K. (2023). Supervised dimensionality reduction for multiple imputation by chained equations. *arXiv preprint*. https://doi.org/10.48550/arXiv.2309.01608

Costantini, E., Lang, K. M., Sijtsma, K., & Reeskens, T. (2024). Solving the many-variables problem in MICE with principal component regression. *Behavior Research Methods*, *56*(3), 1715–1737. https://doi.org/10.3758/s13428-023-02117-1

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, *34*(2), 187–202. https://doi.org/10.1111/j.2517-6161.1972.tb00899.x

Cramér, H. (1946). *Mathematical methods of statistics*. Princeton University Press. http://www.jstor.org/stable/j.ctt1bpm9r4

Croux, C., & Ruiz-Gazen, A. (2005). High breakdown estimators for principal components: The projection-pursuit approach revisited. *Journal of Multivariate Analysis*, *95*(1), 206–226. https://doi.org/10.1016/j.jmva.2004.08.002

de Leeuw, E. D., Hox, J. J., & Huisman, M. (2003). Prevention and treatment of item nonresponse. *Journal of Official Statistics*, *19*, 153–176.

de Jong, S., & Kiers, H. A. (1992). Principal covariates regression: Part I. theory. *Chemometrics and Intelligent Laboratory Systems*, *14*(1-3), 155–164. https://doi.org/10.1016/0169-7439(92)80100-I

Dempster, A. P., Schatzoff, M., & Wermuth, N. (1977). A simulation study of alternatives to ordinary least squares. *Journal of the American Statistical Association*, *72*(357), 77–91. https://doi.org/10.2307/2286909

Deng, Y., Chang, C., Ido, M. S., & Long, Q. (2016). Multiple imputation for general missing data patterns in the presence of high-dimensional data. *Scientific Reports*, *6*(1), 21689. https://doi.org/10.1038/srep21689

de Schipper, N. C., & Van Deun, K. (2019). Revealing the joint mechanisms in traditional data linked with big data. *Zeitschrift für Psychologie*, *226*(4), 212–231. https://doi.org/10.1027/2151-2604/a000341

Ding, B., & Gentleman, R. (2005). Classification using generalized partial least squares. *Journal of Computational and Graphical Statistics*, *14*(2), 280–298. https://doi.org/10.1198/106186005X47697

Doove, L. L., van Buuren, S., & Dusseldorp, E. (2014). Recursive partitioning for missing data imputation in the presence of interaction effects. *Computational Statistics & Data Analysis*, *72*, 92–104. https://doi.org/10.1016/j.csda.2013.10.025

Eddelbuettel, D. (2013). *Seamless R and C++ integration with Rcpp*. Springer. https://doi.org/10.1007/978-1-4614-6868-4

Eddelbuettel, D., & Balamuta, J. J. (2018). Extending r with c++: A brief introduction to rcpp. *The American Statistician*, *72*(1), 28–36. https://doi.org/10.1080/00031305.2017.1375990

Eekhout, I., de Vet, H. C., de Boer, M. R., Twisk, J. W., & Heymans, M. W. (2018). Passive imputation and parcel summaries are both valid to handle missing items in studies with many multi-item scales. *Statistical Methods in Medical Research*, *27*(4), 1128–1140. https://doi.org/10.1177/0962280216654511

Eekhout, I., de Vet, H. C., Twisk, J. W., Brand, J. P., de Boer, M. R., & Heymans, M. W. (2014). Missing data in a multi-item instrument were best handled by multiple imputation at the item score level. *Journal of Clinical Epidemiology*, *67*(3), 335–342. https://doi.org/10.1016/j.jclinepi.2013.09.009

Efroymson, M. (1966). Stepwise regression–a backward and forward look. *Eastern Regional Meetings of the Institute of Mathematical Statistics*, 27–29.

Enders, C. K. (2010). *Applied missing data analysis*. The Guilford Press.

Enders, C. K., Mistler, S. A., & Keller, B. T. (2016). Multilevel multiple imputation: A review and evaluation of joint modeling and chained equations imputation. *Psychological Methods*, *21*(2), 222–240. https://doi.org/10.1037/met0000063

Erikson, R., Goldthorpe, J. H., & Portocarero, L. (1979). Intergenerational class mobility in three western european societies: England, france and sweden. *The British Journal of Sociology*, *30*(4), 415–441. https://doi.org/10.2307/589632

Estefan, L. F., Vivolo-Kantor, A. M., Niolon, P. H., Le, V. D., Tracy, A. J., Little, T. D., DeGue, S., Latzman, N. E., Tharp, A., Lang, K. M., et al. (2021). Effects of the dating matters® comprehensive prevention model on health- and delinquency-related risk behaviors in middle school youth: A cluster-randomized controlled trial. *Prevention Science*, *22*(2), 163–174. https://doi.org/10.1007/s11121-020-01114-6

EVS. (2020a). European values study 2017: Integrated dataset (evs 2017). https://doi.org/10.4232/1.13511

EVS. (2020b). EVS bibliography [Retrieved September 30, 2020]. https://europeanvaluesstudy.eu/education-dissemination-publications/evs-publications/publications/

Filmer, D., & Pritchett, L. H. (2001). Estimating wealth effects without expenditure data—or tears: An application to educational enrollments in states of india. *Demography*, *38*(1), 115–132. https://doi.org/10.1353/dem.2001.0003

Frank, L. E., & Friedman, J. H. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, *35*(2), 109–135. https://doi.org/10.1080/00401706.1993.10485033

Gillespie, C., & Lovelace, R. (2016). *Efficient R programming: A practical guide to smarter programming*. O'Reilly Media, Inc.

Glynn, R. J., Laird, N. M., & Rubin, D. B. (2000). Selection modeling versus mixture modeling with nonignorable nonresponse. In *Drawing inferences from self-selected samples* (1st ed., pp. 115–142). Routledge. https://doi.org/10.4324/9780203774786

Gottschall, A. C., West, S. G., & Enders, C. K. (2012). A comparison of item-level and scale-level multiple imputation for questionnaire batteries. *Multivariate Behavioral Research*, *47*(1), 1–25. https://doi.org/10.1080/00273171.2012.640589

Graham, J. W. (2012). *Missing data: Analysis and design*. Springer. https://doi.org/10.1007/978-1-4614-4018-5

Groves, R. M., Fowler Jr, F. J., Couper, M. P., Lepkowski, J. M., Singer, E., & Tourangeau, R. (2009). *Survey methodology* (2nd ed.). John Wiley & Sons.

Guadagnoli, E., & Cleary, P. D. (1992). Age-related item nonresponse in surveys of recently discharged patients. *Journal of Gerontology*, *47*(3), 206–212. https://doi.org/10.1093/geronj/47.3.P206

Guttman, L. (1954). Some necessary conditions for common-factor analysis. *Psychometrika*, *19*(2), 149–161. https://doi.org/10.1007/BF02289162

Haerpfer, C., Inglehart, R., Moreno, A., Welzel, C., Kizilova, K., J., D.-M., Lagos, M., Norris, P., Ponarin, E., Puranen, B., & et al. (2020). World values survey: Round seven – country-pooled datafile. madrid, spain & vienna, austria. https://doi.org/10.14281/18241.1

Halko, N., Martinsson, P.-G., & Tropp, J. A. (2011). Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, *53*(2), 217–288. https://doi.org/10.1137/090771806

Hans, C. (2009). Bayesian lasso regression. *Biometrika*, *96*(4), 835–845. https://doi.org/10.1093/biomet/asp047

Hans, C. (2010a). *Blasso: Mcmc for bayesian lasso regression model* [R package version 0.3]. http://www.stat.osu.edu/~hans/

Hans, C. (2010b). Model uncertainty and variable selection in bayesian lasso regression. *Statistics and Computing*, *20*(2), 221–229. https://doi.org/10.1007/s11222-009-9160-9

Hardt, J., Herke, M., & Leonhart, R. (2012). Auxiliary variables in multiple imputation in regression with missing x: A warning against including too many in small sample research. *BMC Medical Research Methodology*, *12*(1), 1–13. https://doi.org/10.1186/1471-2288-12-184

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer. https://doi.org/10.1007/978-0-387-21606-5

Hastie, T., Tibshirani, R., & Tibshirani, R. J. (2017). Extended comparisons of best subset selection, forward stepwise selection, and the lasso. *arXiv preprint*. https://doi.org/10.48550/arXiv.1707.08692

Heckman, J. J. (1976). The common structure of statistical models of truncation, sample selection and limited dependent variables and a simple estimator for such models. *Annals of Economic and Social Measurement*, *5*(4), 475–492.

Heckman, J. J. (1979). Sample selection bias as a specification error. *Econometrica: Journal of the Econometric Society*, *47*(1), 153–161. https://doi.org/10.2307/1912352

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, *12*(1), 55–67. https://doi.org/10.1080/00401706.1970.10488634

Howard, W. J., Rhemtulla, M., & Little, T. D. (2015). Using principal components as auxiliary variables in missing data estimation. *Multivariate Behavioral Research*, *50*(3), 285–299. https://doi.org/10.1080/00273171.2014.999267

Hubert, M., Rousseeuw, P., & Verdonck, T. (2009). Robust PCA for skewed data and its outlier map. *Computational Statistics & Data Analysis*, *53*(6), 2264–2274. https://doi.org/10.1016/j.csda.2008.05.027

IBM Corp. (2020). *IBM SPSS missing values v27* (Version 27.0). Armonk, NY: IBM Corp. https://www.ibm.com/docs/en/SSLVMB_27.0.0/pdf/en/IBM_SPSS_Missing_Values.pdf

Immerzeel, T., Coffé, H., & Van der Lippe, T. (2015). Explaining the gender gap in radical right voting: A cross-national investigation in 12 western european countries. *Comparative European Politics*, *13*(2), 263–286. https://doi.org/10.1057/cep.2013.20

Inglehart, R. (1997). *Modernization and postmodernization: Cultural, economic, and political change in 43 societies*. Princeton University Press. https://doi.org/10.2307/j.ctv10vm2ns

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning: With applications in R* (2nd ed.). Springer. https://doi.org/10.1007/978-1-0716-1418-1

Jolliffe, I. T. (2002). *Principal component analysis* (2nd ed.). Springer. https://doi.org/10.1007/b98835

Kaiser, H. F. (1960). The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, *20*(1), 141–151. https://doi.org/10.1177/001316446002000116

Kennickell, A. B. (2017). Multiple imputation in the survey of consumer finances. *International Journal of Computer Applications*, *33*(1), 143–151. https://doi.org/10.3233/SJI-160278

Kiers, H. A. L. (1991). Simple structure in component analysis techniques for mixtures of qualitative and quantitative variables. *Psychometrika*, *56*(2), 197–212. https://doi.org/10.1007/BF02294458

Kolenikov, S., & Angeles, G. (2009). Socioeconomic status measurement with discrete proxy variables: Is principal component analysis a reliable answer? *Review of Income and Wealth*, *55*(1), 128–165. https://doi.org/10.1111/j.1475-4991.2008.00309.x

Köneke, V. (2014). Trust increases euthanasia acceptance: A multilevel analysis using the european values study. *BMC Medical Ethics*, *15*(1), 1–17. https://doi.org/10.1186/1472-6939-15-86

Krämer, N., & Sugiyama, M. (2011). The degrees of freedom of partial least squares regression. *Journal of the American Statistical Association*, *106*(494), 697–705. https://doi.org/10.1198/jasa.2011.tm10107

Lang, K. M., Little, T. D., & PcAux Development Team. (2018). *Pcaux: Automatically extract auxiliary features for simple, principled missing data analysis* [R package version 0.0.0.9013]. https://github.com/PcAux-Package/PcAux

Ley, E., & Steel, M. F. (2009). On the effect of prior assumptions in bayesian model averaging with applications to growth regression. *Journal of Applied Econometrics*, *24*(4), 651–674. https://doi.org/10.1002/jae.1057

Little, N., Roderick J. A.and Zhang. (2011). Subsample ignorable likelihood for regression analysis with missing data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *60*(4), 591–605. https://doi.org/10.1111/j.1467-9876.2011.00763.x

Little, R. J. A., & Rubin, D. B. (2002). *Statistical analysis with missing data* (2nd ed.). John Wiley & Sons. https://doi.org/10.1002/9781119013563

Little, R. J. A. (1988). Missing-data adjustments in large surveys. *Journal of Business & Economic Statistics*, *6*(3), 287–296. https://doi.org/10.1080/07350015.1988.10509663

Little, R. J. A. (1993). Pattern-mixture models for multivariate incomplete data. *Journal of the American Statistical Association*, *88*(421), 125–134. https://doi.org/10.1080/01621459.1993.10594302

Little, T. D., Jorgensen, T. D., Lang, K. M., & Moore, E. W. G. (2013). On the joys of missing data. *Journal of Pediatric Psychology*, *39*(2), 1–12. https://doi.org/10.1093/jpepsy/jsto48

Lüdtke, O., Robitzsch, A., & Grund, S. (2017). Multiple imputation of missing data in multilevel designs: A comparison of different strategies. *Psychological Methods*, *22*(1), 141–165. https://doi.org/10.1037/met0000096

Luijkx, R., Jónsdóttir, G. A., Gummer, T., Ernst Stähli, M., Frederiksen, M., Ketola, K., Reeskens, T., Brislinger, E., Christmann, P., Gunnarsson, S. Þ., et al. (2021). The european values study 2017: On the way to the future using mixed-modes. *European Sociological Review*, *37*(2), 330–346. https://doi.org/10.1093/esr/jcaa049

LWS. (2020). Luxembourg wealth study database. https://www.lisdatacenter.org/

Mainzer, R., Apajee, J., Nguyen, C. D., Carlin, J. B., & Lee, K. J. (2021). A comparison of multiple imputation strategies for handling missing data in multi-item scales: Guidance for longitudinal studies. *Statistics in Medicine*, *40*(21), 4660–4674. https://doi.org/10.1002/sim.9088

Massey, D. S., & Tourangeau, R. (2013). Where do we go from here? Nonresponse and social measurement. *The ANNALS of the American Academy of Political and Social Science*, *645*(1), 222–236. https://doi.org/10.1177/0002716212464191

McCrae, R. R., & Costa Jr, P. T. (1983). Joint factors in self-reports and ratings: Neuroticism, extraversion and openness to experience. *Personality and Individual Differences*, *4*(3), 245–255. https://doi.org/10.1016/0191-8869(83)90146-0

McGonagle, K. A., Schoeni, R. F., Sastry, N., & Freedman, V. A. (2012). The panel study of income dynamics: Overview, recent innovations, and potential for life course research. *Longitudinal and Life Course Studies*, *3*(2), 1–21. https://doi.org/10.14301/llcs.v3i2.188

Meng, X.-L. (1994). Multiple-imputation inferences with uncongenial sources of input. *Statistical Science*, *9*(4), 538–558. https://www.jstor.org/stable/2246252

Meyer, B. D., Mok, W. K., & Sullivan, J. X. (2015). Household surveys in crisis. *Journal of Economic Perspectives*, *29*(4), 199–226. https://doi.org/10.1257/jep.29.4.199

Mustillo, S. (2012). The effects of auxiliary variables on coefficient bias and efficiency in multiple imputation. *Sociological Methods & Research*, *41*(2), 335–361. https://doi.org/10.1177/0049124112452392

Mustillo, S., & Kwon, S. (2015). Auxiliary variables in multiple imputation when data are missing not at random. *The Journal of Mathematical Sociology*, *39*(2), 73–91. https://doi.org/10.1080/0022250X.2013.877898

Muthén, B., Kaplan, D., & Hollis, M. (1987). On structural equation modeling with data that are not missing completely at random. *Psychometrika*, *52*(3), 431–462. https://doi.org/10.1007/BF02294365

Nagelkerke, N. J. (1991). A note on a general definition of the coefficient of determination. *Biometrika*, *78*(3), 691–692. https://doi.org/10.1093/biomet/78.3.691

Nelder, J. A., & Wedderburn, R. W. M. (1972). Generalized linear models. *Journal of the Royal Statistical Society Series A: Statistics in Society*, *135*(3), 370–384. https://doi.org/10.2307/2344614

Niolon, P. H., Vivolo-Kantor, A. M., Tracy, A. J., Latzman, N. E., Little, T. D., DeGue, S., Lang, K. M., Estefan, L. F., Ghazarian, S. R., McIntosh, W. L. K., et al. (2019). An RCT of dating matters: Effects on teen dating violence and relationship behaviors. *American Journal of Preventive Medicine*, *57*(1), 13–23. https://doi.org/10.1016/j.amepre.2019.02.022

Oberman, H. I., & Vink, G. (2023). Towards a standardized evaluation of imputation methodology. *Biometrical Journal*, *66*(1), 1–12. https://doi.org/10.1002/bimj.202200107

Park, S., Ceulemans, E., & Van Deun, K. (2021). Sparse common and distinctive covariates regression. *Journal of Chemometrics*, *35*(2), 1–23. https://doi.org/10.1002/cem.3270

Park, S., Ceulemans, E., & Van Deun, K. (2023). Logistic regression with sparse common and distinctive. *Behavior Research Methods*, 1–32. https://doi.org/10.3758/s13428-022-02011-2

Park, T., & Casella, G. (2008). The bayesian lasso. *Journal of the American Statistical Association*, *103*(482), 681–686. https://doi.org/10.1198/016214508000000337

Peugh, J. L., & Enders, C. K. (2004). Missing data in educational research: A review of reporting practices and suggestions for improvement. *Review of Educational Research*, *74*(4), 525–556. https://doi.org/10.3102/00346543074004525

Prentice, R. L., & Gloeckler, L. A. (1978). Regression analysis of grouped survival data with application to breast cancer data. *Biometrics*, 57–67. https://doi.org/10.2307/2529588

R Core Team. (2020). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/

R Core Team. (2023). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/

Raghunathan, T. E. (2021). Synthetic data. *Annual Review of Statistics and its Application*, *8*, 129–140. https://doi.org/10.1146/annurev-statistics-040720-031848

Raghunathan, T. E., Lepkowski, J. M., Van Hoewyk, J., & Solenberger, P. (2001). A multivariate technique for multiply imputing missing values using a sequence of regression models. *Survey Methodology*, *27*(1), 85–96.

Raghunathan, T. E., Solenberger, P. W., & Van Hoewyk, J. (2002). Iveware: Imputation and variance estimation software. *Ann Arbor, MI: Survey Methodology Program, Survey Research Center, Institute for Social Research, University of Michigan*.

Raíche, G. (2010). *An R package for parallel analysis and non graphical solutions to the Cattell scree test* [R package version 2.3.3.1.]. https://CRAN.R-project.org/package=nFactors

Raíche, G., Walls, T. A., Magis, D., Riopel, M., & Blais, J.-G. (2013). Non-graphical solutions for Cattell's scree test. *Methodology*, *9*(1), 23–29. https://doi.org/10.1027/1614-2241/a000051

Reiss, P. T., & Ogden, R. T. (2007). Functional principal component regression and functional partial least squares. *Journal of the American Statistical Association*, *102*(479), 984–996. https://doi.org/10.1198/016214507000000527

Reiter, J. P., Raghunathan, T. E., & Kinney, S. K. (2006). The importance of modeling the sampling design in multiple imputation for missing data. *Survey Methodology*, *32*(2), 143–149.

Robitzsch, A., & Grund, S. (2022). *Miceadds: Some additional multiple imputation functions, especially for 'mice'* [R package version 3.15-21]. https://CRAN.R-project.org/package=miceadds

Robitzsch, A., Pham, G., & Yanagida, T. (2016). Fehlende daten und plausible values. In *Large-scale assessment mit R: Methodische grundlagen der österreichischen bildungsstandardüberprüfung* (pp. 259–293). facultas.

Rosipal, R., Girolami, M., Trejo, L. J., & Cichocki, A. (2001). Kernel PCA for feature extraction and de-noising in nonlinear regression. *Neural Computing & Applications*, *10*(3), 231–243. https://doi.org/10.1007/s521-001-8051-z

Rubin, D. B. (1976). Inference and missing data. *Biometrika*, *63*(3), 581–592. https://doi.org/10.1093/biomet/63.3.581

Rubin, D. B. (1986). Statistical matching using file concatenation with adjusted weights and multiple imputations. *Journal of Business & Economic Statistics*, *4*(1), 87–94. https://doi.org/10.1080/07350015.1986.10509497

Rubin, D. B. (1987). *Multiple imputation for nonresponse in surveys*. John Wiley & Sons. https://doi.org/10.1002/9780470316696

Rubin, D. B. (1996). Multiple imputation after 18+ years. *Journal of the American Statistical Association*, *91*(434), 473–489. https://doi.org/10.1080/01621459.1996.10476908

Rubin, D. B., Stern, H. S., & Vehovar, V. (1995). Handling “don’t know” survey responses: The case of the slovenian plebiscite. *Journal of the American Statistical Association*, *90*(431), 822–828. https://doi.org/10.1080/01621459.1995.10476580

Schafer, J. L. (1997). *Analysis of incomplete multivariate data* (1st ed.). Chapman & Hall/CRC. https://doi.org/10.1201/9780367803025

Schafer, J. L., & Graham, J. W. (2002). Missing data: Our view of state of the art. *Psychological Methods*, *7*(2), 147–177. https://doi.org/10.1037//1082-989X.7.2.147

Scherpenzeel, A. C., & Das, M. (2018). “True” longitudinal and probability-based internet panels: Evidence from the Netherlands. In *Social and behavioral research and the internet* (pp. 77–104). Routledge. https://doi.org/10.4324/9780203844922

Schouten, R. M., & Vink, G. (2021). The dance of the mechanisms: How observed information influences the validity of missingness assumptions. *Sociological Methods & Research*, *50*(3), 1243–1258. https://doi.org/10.1177/0049124118799376

Scott, J. G., & Berger, J. O. (2010). Bayes and empirical-bayes multiplicity adjustment in the variable-selection problem. *The Annals of Statistics*, *38*(5), 2587–2619. https://doi.org/10.1214/10-AOS792

Shah, A. D. (2018). *CALIBERrfimpute: Imputation in MICE using random forest* [R package version 1.0-1].

Shah, A. D., Bartlett, J. W., Carpenter, J., Nicholas, O., & Hemingway, H. (2014). Comparison of random forest and parametric imputation models for imputing missing data using MICE: A CALIBER study. *American Journal of Epidemiology*, *179*(6), 764–774. https://doi.org/10.1093/aje/kwt312

Shen, D., Shen, H., & Marron, J. S. (2016). A general framework for consistency of principal component analysis. *Journal of Machine Learning Research*, *17*(150), 1–34.

Smith, A. F., & Spiegelhalter, D. J. (1980). Bayes factors and choice criteria for linear models. *Journal of the Royal Statistical Society: Series B (Methodological)*, *42*(2), 213–220. https://doi.org/10.1111/j.2517-6161.1980.tb01122.x

Song, J., & Belin, T. R. (2004). Imputation for incomplete high-dimensional multivariate normal data using a common factor model. *Statistics in Medicine*, *23*(18), 2827–2843. https://doi.org/10.1002/sim.1867

Spearman, C. (1904). The proof and measurement of association between two things. *The American Journal of Psychology*, *15*(1), 72–101. https://doi.org/10.2307/1422689

StataCorp. (2013). *Stata 17 multiple-imputation reference manual*. College Station, TX: StataCorp LLC, A Stata Press Publication College Station, TX.

StataCorp. (2023). *Stata 18 user's guide*. College Station, TX: StataCorp LLC.

Stone, M., & Brooks, R. J. (1990). Continuum regression: Cross-validated sequentially constructed prediction embracing ordinary least squares, partial least squares and principal components regression. *Journal of the Royal Statistical Society: Series B (Methodological)*, *52*(2), 237–258. https://doi.org/10.1111/j.2517-6161.1990.tb01786.x

Tenenhaus, M., & Young, F. W. (1985). An analysis and synthesis of multiple correspondence analysis, optimal scaling, dual scaling, homogeneity analysis and other methods for quantifying categorical multivariate data. *Psychometrika*, *50*(1), 91–119. https://doi.org/10.1007/BF02294151

Tharp, A. T. (2012). Dating matters™: The next generation of teen dating violence prevention. *Prevention Science*, *13*(4), 398–401. https://doi.org/10.1007/s11121-012-0307-0

The pandas development team. (2020, February). *Pandas-dev/pandas: Pandas* (Version latest). Zenodo. https://doi.org/10.5281/zenodo.3509134

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Timmerman, M. E. (2006). Multilevel component analysis. *British Journal of Mathematical and Statistical Psychology*, *59*(2), 301–320. https://doi.org/10.1348/000711005X67599

van Buuren, S. (2007). Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical Methods in Medical Research*, *16*(3), 219–242. https://doi.org/10.1177/0962280206074463

van Buuren, S. (2010). Item imputation without specifying scale structure. *Methodology*, *6*(1), 31–36. https://doi.org/10.1027/1614-2241/a000004

van Buuren, S. (2011). Multiple imputation of multilevel data. In J. Hox & J. K. Roberts (Eds.), *Handbook of advanced multilevel analysis* (pp. 173–196). Routledge. https://doi.org/10.4324/9780203848852

van Buuren, S. (2018). *Flexible imputation of missing data* (2nd ed.). Chapman; Hall/CRC. https://doi.org/10.1201/9780429492259

van Buuren, S., Boshuizen, H. C., & Knook, D. L. (1999). Multiple imputation of missing blood pressure covariates in survival analysis. *Statistics in Medicine*, *18*(6), 681–694. https://doi.org/10.1002/(SICI)1097-0258(19990330)18:6%3C681::AID-SIM71%3E3.0.CO;2-R

van Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in R. *Journal of Statistical Software*, *45*(3), 1–67. https://doi.org/10.18637/jss.v045.i03

van Buuren, S., & Oudshoorn, C. G. M. (2000). *Multivariate imputation by chained equations: Mice v1.0 user's manual* (tech. rep. No. PG/VGZ/00.038). TNO Prevention and Health.

Van Deun, K., Smilde, A. K., Van Der Werf, M. J., Kiers, H. A., & Van Mechelen, I. (2009). A structured overview of simultaneous component based data integration. *BMC Bioinformatics*, *10*, 1–15. https://doi.org/10.1186/1471-2105-10-246

Van Deun, K., Wilderjans, T. F., van den Berg, R. A., Antoniadis, A., & Van Mechelen, I. (2011). A flexible framework for sparse simultaneous component based data integration. *BMC Bioinformatics*, *12*(1), 1–17. https://doi.org/10.1186/1471-2105-12-448 van Ginkel, J. R., Sijtsma, K., van der Ark, L. A., & Vermunt, J. K. (2010). Incidence of missing item scores in personality measurement, and simple item-score imputation. *Methodology*, *6*(1), 17–30. https://doi.org/10.1027/1614-2241/a000003

Vervloet, M., Van Deun, K., Van den Noortgate, W., & Ceulemans, E. (2013). On the selection of the weighting parameter value in principal covariates regression. *Chemometrics and Intelligent Laboratory Systems*, *123*, 36–43. https://doi.org/10.1016/j.chemolab.2013.02.005

Vervloet, M., Van Deun, K., Van den Noortgate, W., & Ceulemans, E. (2016). Model selection in principal covariates regression. *Chemometrics and Intelligent Laboratory Systems*, *151*, 26–33. https://doi.org/10.1016/j.chemolab.2015.12.004

Vivolo-Kantor, A. M., Niolon, P. H., Estefan, L. F., Le, V. D., Tracy, A. J., Latzman, N. E., Little, T. D., Lang, K. M., DeGue, S., & Tharp, A. T. (2021). Middle school effects of the dating matters® comprehensive teen dating violence prevention model on physical violence, bullying, and cyberbullying: A cluster-randomized controlled trial. *Prevention Science*, *22*(2), 151–161. https://doi.org/10.1007/s11121-019-01071-9

Volker, T. B., & Vink, G. (2021). Anonymiced shareable data: Using mice to create and analyze multiply imputed synthetic datasets. *Psych*, *3*(4), 703–716. https://doi.org/10.3390/psych3040045

von Hippel, P. T. (2009). How to impute interactions, squares, and other transformed variables. *Sociological Methodology*, *39*(1), 265–291. https://doi.org/10.1111/j.1467-9531.2009.01215.x

von Hippel, P. T., & Lynch, J. (2013). Efficiency gains from using auxiliary variables in imputation. *arXiv preprint*. https://doi.org/10.48550/arXiv.1311.5249

White, I. R., & Carlin, J. B. (2010). Bias and efficiency of multiple imputation compared with complete-case analysis for missing covariate values. *Statistics in Medicine*, *29*(28), 2920–2931. https://doi.org/10.1002/sim.3944

White, I. R., Royston, P., & Wood, A. M. (2011). Multiple imputation using chained equations: Issues and guidance for practice. *Statistics in Medicine*, *30*(4), 377–399. https://doi.org/10.1002/sim.4067

Williams, D., & Brick, J. M. (2018). Trends in U.S. face-to-face household survey nonresponse and level of effort. *Journal of Survey Statistics and Methodology*, *6*(2). https://doi.org/10.1093/jssam/smx019

Wold, H. (1975). Path models with latent variables: The NIPALS approach. In *Quantitative sociology* (pp. 307–357). Elsevier. https://doi.org/10.1016/B978-0-12-103950-9.50017-4

Zhao, Y., & Long, Q. (2016). Multiple imputation in the presence of high-dimensional data. *Statistical Methods in Medical Research*, *25*(5), 2021–2035. https://doi.org/10.1177/0962280213511027

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, *101*(476), 1418–1429. https://doi.org/10.1198/016214506000000735

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, *67*(2), 301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x

Zou, H., Hastie, T., & Tibshirani, R. (2006). Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, *15*(2), 265–286. https://doi.org/https://doi.org/10.1198/106186006X113430

Zwick, W. R., & Velicer, W. F. (1986). Comparison of five rules for determining the number of components to retain. *Psychological Bulletin*, *99*(3), 432. https://doi.org/10.1037/0033-2909.99.3.432

Summary

Social scientists commonly analyze data collected through surveys, which often suffer from nonresponse. Survey respondents may fail or refuse to provide answers to certain questions, which creates missing values in the data. Missing values can disrupt even the most fundamental analyses. For example, the correlation between two variables with missing values cannot be computed without deciding how to handle such values. To address the missingness, a researcher might decide to compute the correlation using only the cases that are completely observed on both variables. Complete case analysis is intuitive and easy to use but it wastes a portion of the observed data and can bias estimation. Multiple Imputation (MI) is an alternative state-of-the-art method that replaces the missing values in the original data with plausible ones, multiple times, leading to the definition of multiple imputed versions of the original data. After obtaining imputations, the same analysis model can be estimated on all imputed versions of the data. The multiple resulting parameter estimates can be averaged to produce a single interpretable analysis. MI avoids wasting observed data and, if the right assumptions are met, leads to unbiased parameter estimates and valid inferential conclusions.

In this dissertation, we focused on a specific implementation of MI known as Multivariate Imputation by Chained Equations (MICE). MICE obtains imputations for multivariate missing data through a collection of univariate regression models, known as imputation models, which define a separate univariate conditional distribution for every variable that needs to be imputed. By doing so, MICE can impute data consisting of a mix of continuous, binary, and categorical variables, making it particularly suited to address problems of missing values in social science data. However, the need to conditionally specify an imputation model for every variable under imputation also poses the challenge of defining a set of predictors for every model. Every variable in the data is a possible predictor for the imputation model of every variable with missing values. This can make the selection of the predictors a daunting task, especially when data contain hundreds of variables.

This dissertation proposed an approach to MI that uses a Generalized version of Supervised Principal Component Regression as a univariate imputation model in MICE (MI-GSPCR) to simplify the choice of imputation predictors when analyzing

social science data with hundreds of variables. Principal Component Regression (PCR) is a dimensionality reduction approach to regression analysis that replaces the predictors in a regression model with their Principal Components (PCs), which are constructed variables that summarize the information contained in the original predictors and can stabilize the estimation of regression models. Using PCR as an imputation model in MICE allows the analyst to provide all variables in the data as predictors in the imputation model, and thus facilitates the use of MICE by removing the need to choose predictors. Supervision ensures that the PCs computed based on all the potential predictors are relevant for the prediction of each variable under imputation. Generalization allows the imputation of both continuous and categorical data, based on both continuous and categorical predictors.

We arrived at the MI-GSPCR formulation by building on the results of each chapter. In Chapter 2, we identified the use of Principal Component Analysis (PCA) as a promising method to reduce the complexity of the variable selection task. We found that replacing the many potential predictors in the imputation models with PCs resulted in minimal bias and acceptable confidence interval coverage. In this chapter, PCA was computed before imputation to reduce the dimensionality of a set of fully observed potential predictors. This created a small set of summary variables that were used in place of the many potential predictors in the imputation models of a subsequent run of MICE. However, this use of PCA is limiting because it requires a large number of potential predictors with no missing values, a rarity in social scientific data. In Chapter 3, we addressed this limitation by exploring different ways of using PCA as part of the imputation procedure. We proposed using PCR as a univariate imputation model in MICE, which implied estimating new PCs for every variable under imputation, at every iteration of the MICE algorithm. Through a simulation study, we showed that this approach resulted in low bias and good confidence interval coverage properties. We also explored how these results were affected by the use of different numbers of PCs and discovered that current recommendations were inadequate when applied to data with a latent structure. Our simulation study suggested that the number of PCs should be at least as large as the number of latent variables in the data-generating model, a result which we revisited in Chapter 6. In Chapter 4, we explored how the addition of supervision could improve upon the use of classical unsupervised PCA. In this context, supervision means using the variables that are under imputation to help compute PCs that can be good predictors. We designed a simulation study to compare different approaches to supervised dimensionality reduction as imputation models for MICE. We found that all forms of supervision obtained better imputations using fewer PCs than the original unsupervised PCA-based approach. Finally, in Chapter 5, we adapted the use of supervised PCR to allow the imputation of data consisting of a mix of continuous and categorical data, a fundamental characteristic of survey data. We also developed software to perform the imputation and demonstrated its use on a real survey data set. Altogether, this dissertation presents an extensive study of how PCs can be used to simplify the selection of predictors for the imputation models in the MICE approach to MI.

Samenvatting

Sociale wetenschappers analyseren vaak data die via enquêtes zijn verzameld. Een veelvoorkomend probleem bij deze enquêtes is non-respons. Het komt voor dat respondenten bepaalde vragen niet beantwoorden of weigeren te antwoorden, wat leidt tot ontbrekende waarden in de data. Zelfs bij de meest fundamentele analyses kunnen deze ontbrekende waarden verstorend werken. Zo kan bijvoorbeeld de correlatie tussen twee variabelen met ontbrekende waarden niet worden berekend zonder te beslissen hoe met dergelijke waarden moet worden omgegaan. Een onderzoeker zou kunnen besluiten de correlatie alleen te berekenen met observaties die volledig zijn waargenomen voor beide variabelen. Deze analyse van complete data is intuïtief en eenvoudig toe te passen, maar gaat ten koste van een deel van de waargenomen data en kan de schatting vertekenen. Multiple Imputation (MI) is een alternatieve, geavanceerde methode waarbij de ontbrekende waarden in de originele data meerdere keren worden vervangen door plausibele waarden. Dit resulteert in meerdere geïmputeerde versies van de oorspronkelijke data. Na het verkrijgen van de imputaties kan hetzelfde analysemodel worden toegepast op alle geïmputeerde versies van de data. Door het gemiddelde van deze verschillende parameterschattingen te berekenen, ontstaat een enkele interpreteerbare analyse. MI vermijdt het verspillen van geobserveerde data en leidt, mits aan de juiste aannames wordt voldaan, tot zuivere parameterschattingen en geldige conclusies.

Dit proefschrift richt zich op een specifieke implementatie van MI, bekend als Multivariate Imputation by Chained Equations (MICE). MICE verkrijgt imputaties voor multivariate ontbrekende data door middel van een reeks univariate regressiemodellen, ook wel imputatiemodellen genoemd. Deze modellen definiëren een aparte univariate conditionele verdeling voor elke te imputeren variabele. Hierdoor kan MICE data imputeren die bestaan uit een mix van continue, binaire en categorische variabelen, wat het bijzonder geschikt maakt voor het omgaan met ontbrekende waarden in sociaalwetenschappelijke data. De noodzaak om voor elke te imputeren variabele een conditioneel imputatiemodel te specificeren, brengt echter een uitdaging met zich mee: het definiëren van een set predictoren voor elk model. Elke variabele in de dataset is een potentiële predictor voor het imputatiemodel van elke variabele met ontbrekende waarden. Dit kan de selectie van predictoren bemoeilijken, vooral bij datasets met honderden variabelen.

In dit proefschrift stellen we een MI-benadering voor die een Gegeneraliseerde versie van Supervised Principal Component Regression gebruikt als univariaat imputatiemodel in MICE (MI-GSPCR). Deze methode vereenvoudigt de selectie van imputatiepredictoren bij het analyseren van sociaalwetenschappelijke data met honderden variabelen. We passen Principale Componentenregressie (PCR) toe, een dimensiereductietechniek waarbij predictoren in een regressiemodel worden vervangen door hun principale componenten (PC's). Deze PC's zijn samengestelde variabelen die de informatie in de oorspronkelijke predictoren samenvatten en de schatting van regressiemodellen kunnen stabiliseren. Door PCR als imputatiemodel in MICE te gebruiken, kunnen onderzoekers alle variabelen in de dataset als predictoren in het imputatiemodel opnemen. Dit vergemakkelijkt het gebruik van MICE doordat er geen predictoren hoeven te worden geselecteerd. De supervisie zorgt ervoor dat de PC's, berekend op basis van alle potentiële predictoren, relevant blijven voor het voorspellen van elke te imputeren variabele. De generalisatie maakt het mogelijk om zowel continue als categorische data te imputeren, gebaseerd op zowel continue als categorische predictoren.

We kwamen tot de MI-GSPCR formulering door voort te bouwen op de resultaten van elk hoofdstuk. In hoofdstuk 2 kwam Principale Componentenanalyse (PCA) naar voren als een veelbelovende methode om de complexiteit van variabelenselectie te verminderen. We ontdekten dat het vervangen van de vele potentiële predictoren in de imputatiemodellen door PC's resulteerde in minimale bias en een acceptabele dekking van de betrouwbaarheidsintervallen. In hoofdstuk 2 werd PCA toegepast vóór de imputatie om de dimensionaliteit van een set volledig geobserveerde potentiële predictoren te reduceren. Dit leidde tot een kleine set samenvattende variabelen die werden gebruikt in plaats van de vele potentiële predictoren in de imputatiemodellen van een daaropvolgende MICE-procedure. Deze toepassing van PCA kent echter beperkingen, aangezien het een groot aantal potentiële predictoren zonder ontbrekende waarden vereist—iets wat zelden voorkomt in sociaalwetenschappelijke data.

Hoofdstuk 3 richtte zich op deze beperking door verschillende manieren te onderzoeken om PCA te gebruiken als onderdeel van de imputatieprocedure. We stelden voor om PCR als univariaat imputatiemodel in MICE te gebruiken, waarbij voor elke te imputeren variabele nieuwe PC's worden geschat bij elke iteratie van het MICE-algoritme. Via een simulatiestudie toonden we aan dat deze aanpak resulteerde in een lage bias en goede dekkingsgraad van de betrouwbaarheidsintervallen. We onderzochten ook hoe deze resultaten werden beïnvloed door het gebruik van verschillende aantallen PC's en ontdekten dat de huidige aanbevelingen ontoereikend waren bij toepassing op data met een latente structuur. Onze simulatiestudie suggereerde dat het aantal PC's minstens zo groot moet zijn als het aantal latente variabelen in het datagenererende model, een bevinding die we in hoofdstuk 6 opnieuw hebben bekeken.

In hoofdstuk 4 bestudeerden we hoe de toevoeging van supervisie het gebruik van de klassieke PCA zonder supervisie kan verbeteren. Supervisie betekent hier het gebruik van de te imputeren variabelen om PC's te berekenen die goede predictoren zijn. We ontwierpen een simulatiestudie om verschillende benaderingen van gesuperviseerde dimensiereductie als imputatiemodellen voor MICE te vergelijken. We ontdekten dat alle vormen van supervisie betere imputaties opleverden door minder PC's te gebruiken dan de oorspronkelijke PCA-gebaseerde aanpak zonder supervisie.

Ten slotte hebben we in hoofdstuk 5 het gebruik van gesuperviseerde PCR aangepast om de imputatie mogelijk te maken van data bestaande uit een mix van continue en categorische variabelen, een fundamenteel kenmerk van enquêtedata. We ontwikkelden ook software om de imputatie uit te voeren en demonstreerden het gebruik ervan op een echte enquêtedataset. Al met al biedt dit proefschrift een uitgebreide studie van hoe PC's kunnen worden gebruikt om de selectie van predictoren voor de imputatiemodellen in de MICE-benadering van MI te vereenvoudigen.

Acknowledgments

A big thank-you to my supervisors Kyle, Tim, and Klaas. You helped me become an independent researcher while always providing the support I needed along the way. You taught me how to defend my ideas and face the ups and downs of being a researcher with pride and steadiness. I also want to thank Soogeun for the pivotal conversations that shaped the direction of my research, Mihai for introducing me to the best productivity tools, Anton for showing the importance of being chill about it, and Esther for pulling me through every restless night. Finally, the biggest thank-you goes to my parents who taught me how to respect myself.