Sociological Methods & Research
2020, Vol. 49(?) 1–12

DOI: 10.1177/ToBeAssigned
journals.sagepub.com/home/smr
SAGE

Missing the Point: Convergence of Multiple Imputation using Chained Equations Algorithms

Hanne I. Oberman

Background

Feedback:

- Focus on convergence, not evaluation of MI in general.
- Combine Theoretical Background and Intro.

At some point, any scientist conducting statistical analyses will run into a missing data problem (Allison 2001). Missingness is problematic because statistical inference cannot be performed on incomplete data without employing *ad hoc* solutions (e.g., list-wise deletion), which may yield wildly invalid results (Van Buuren 2018). A popular answer to the ubiquitous problem of missing information is to use the framework of multiple imputation (MI), proposed by Rubin (1987). MI is an iterative algorithmic procedure in which missing datapoints are 'imputed' (i.e. filled in) several times. The variability between imputations is used to reflect how much uncertainty in the inference is introduced by the missingness. Therefore, MI can provide valid inferences despite missing information (**maybe add that this refers to CIs/p-values**).

To obtain valid inferences with MI, the variability between imputations should be properly represented (Rubin 1987; Van Buuren 2018). If this variability is under-estimated, confidence intervals around estimates will be too narrow, which can yield spurious results. Over-estimation of the variance between imputations results in unnecessarily wide confidence intervals, which can be costly because it lowers the statistical power (**maybe add why this is costly**). Since both

of these situations are undesirable, imputations and their variability should be evaluated. Evaluation measures, however, are currently missing or under-developed in MI software, like the world-leading `mice` package (Van Buuren and Groothuis-Oudshoorn 2011) in `R` (R Core Team 2019). The goal of this research project is to develop novel methodology and guidelines for evaluating MI methods. These tools will subsequently be implemented in an interactive evaluation framework for multiple imputation, which will aid applied researchers in drawing valid inference from incomplete datasets.

This note provides the theoretical foundation towards the diagnostic evaluation of multiple imputation algorithms. For reasons of brevity, we only focus on the MI algorithm implemented in `mice` (Van Buuren and Groothuis-Oudshoorn 2011). The convergence properties of this MI algorithm are investigated through model-based simulation.* The results of this simulation study are guidelines for assessing convergence of MI algorithms.

Terminology This note follows notation and conventions of `mice` (Van Buuren and Groothuis-Oudshoorn 2011). Basic familiarity with MI methodology is assumed. For the theoretical foundation of MI, see Rubin (1987). For an accessible and comprehensive introduction to MI from an applied perspective, see e.g. Van Buuren (2018).

Let Y denote an $n \times p$ matrix containing the data values on p variables for all n units in a sample. The collection of observed data values in Y is denoted by Y_{obs}, and will be referred to as 'incomplete' or 'observed' data. The missing part of Y is denoted by Y_{mis}. Which parts of Y are missing is determined by the 'missingness mechanism'.
This note only considers a 'missing completely at random' (MCAR) mechanism, where the probability of being missing is equal for all $n \times p$ cells in Y (Rubin 1987).

Figure 1 provides an overview of the steps involved with MI—from incomplete data, to m completed datasets, to m estimated quantities of interest ($\hat{Q}$s), to a single pooled estimate $\bar{Q}$. This note focuses on the algorithmic properties of the imputation step.

The MI algorithm in `mice` has an iterative nature. For each missing datapoint in Y_{mis}, m 'chains' of potential values are sampled. Only the ultimate sample that each chain lands on is imputed. The number of iterations per chain will be denoted with T, where t varies over the integers $1, 2, \ldots, T$. This is repeated m times. Imputed values are the ultimate samples of a 'chain' of For each imputed value ($t = T$), a s for each missing datapoint in Y_{mis}, . Each of the m chains starts with an initial value, drawn randomly from Y_{obs}. The chains are terminated after a predefined number of iterations. , and subsequently used in the analysis and

*All programming code used in this note is available from github.com/gerkovink/shinyMice.

pooling steps. The collection of samples between the initial value (at $t = 1$) and the imputed value (at $t = T$) will be referred to as an 'imputation chain'.

Theoretical Background There is no scientific consensus on the convergence properties of multiple imputation algorithms (Takahashi 2017). Some default techniques in `mice` might not yield converged states at all (Murray 2018). Therefore, algorithmic convergence should be monitored carefully. The diagnostic evaluation of convergence is preferred over the current practice—visually inspecting imputation chains for signs of non-convergence—since the latter may be challenging to the untrained eye (Van Buuren 2018, ~6.5.2).

The application of convergence diagnostics to MI methods has not been systematically studied (Van Buuren 2018). The measures that are available for diagnosing convergence of iterative algorithmic procedures (Markov chain Monte Carlo methods; e.g., `mice` algorithms or Gibbs samplers) generally evaluate signs of non-convergence (Hoff 2009). Non-convergence may be present as non-stationarity within chains (i.e., trending), or as slow mixing between chains (i.e., no intermingling).

The stationarity component of convergence may be evaluated with autocorrelation (*AC*; Schafer 1997; Gelman et al. 2013), numeric standard error (or 'MC error'; Geweke 1992), and Raftery and Lewis's (1991) procedure to determine the effect of trending within chains.

The mixing component can be assessed with the potential scale reduction factor $\widehat{R}$ (a.k.a. 'Gelman-Rubin statistic'; Gelman and Rubin 1992). With an adapted version of $\widehat{R}$, proposed by Vehtari et al. (2019), we might also evaluate the stationarity component of convergence. This would make $\widehat{R}$ a general convergence diagnostic. The application of $\widehat{R}$ to assess stationarity has not been thoroughly investigated. Therefore, this study employs both $\widehat{R}$ and autocorrelation to investigate convergence, as recommended by (Cowles and Carlin 1996, ~898).

The mixing component of convergence may be evaluated with the potential scale reduction factor, $\widehat{R}$ (a.k.a. 'Gelman-Rubin statistic'; Gelman and Rubin 1992). The stationarity component may be assessed through autocorrelations. An adapted version of $\widehat{R}$ might also be used to diagnose non-stationarity. Other convergence diagnostics are outside of the scope of this study.

All of these methods evaluate the convergence of univariate scalar summaries (e.g., chain means or variances). These convergence diagnostics cannot diagnose convergence of multivariable statistics (i.e., relations between scalar summaries). Van Buuren (2018) proposed to implement multivariable evaluation through eigenvalue decomposition (MacKay and Mac Kay 2003). This method is outside of the scope of the current study (**not anymore!**).

To define $\widehat{R}$, we follow notation by (Vehtari et al. 2019, ~5). Let M be the total number of chains, T the number of iterations per chain, and θ the scalar summary

of interest (e.g., chain mean or chain variance). For each chain ($m = 1, 2, \ldots, M$), we estimate the variance of θ, and average these to obtain within-chain variance W.

$$W = \frac{1}{M}\sum_{m=1}^{M} s_j^2, \text{ where } s_m^2 = \frac{1}{T-1}\sum_{t=1}^{T}\left(\theta^{(tm)} - \bar{\theta}^{(\cdot m)}\right)^2. <!--$$

We then estimate between-chain variance B as the variance of the collection of average θ per chain.

$$B = \frac{T}{M-1}\sum_{m=1}^{M}\left(\bar{\theta}^{(\cdot m)} - \bar{\theta}^{(\cdot\cdot)}\right)^2, \text{ where } \bar{\theta}^{(\cdot m)} = \frac{1}{T}\sum_{t=1}^{T}\theta^{(tm)}, \quad \bar{\theta}^{(\cdot\cdot)} = \frac{1}{M}\sum_{m=1}^{M}\bar{\theta}^{(\cdot m)}. <!--$$

From the between- and within-chain variances we compute a weighted average, $\widehat{\text{var}}^{+}$, which over-estimates the total variance of θ. $\widehat{R}$ is then obtained as a ratio between the over-estimated total variance and the within-chain variance:

$$\widehat{R} = \sqrt{\frac{\widehat{\text{var}}^{+}(\theta|y)}{W}}, \text{ where } \widehat{\text{var}}^{+}(\theta|y) = \frac{N-1}{N}W + \frac{1}{N}B.$$

We can interpret $\widehat{R}$ as potential scale reduction factor since it indicates by how much the variance of θ could be shrunken down if an infinite number of iterations per chain would be run (Gelman and Rubin 1992). This interpretation assumes that chains are 'over-dispersed' at $t = 1$, and reach convergence as $T \to \infty$. Over-dispersion implies that the initial values of the chains are 'far away' from the target distribution and each other. When all chains sample independent of their initial values, the mixing component of convergence is satisfied, and $\widehat{R}$-values will be close to one. High $\widehat{R}$-values thus indicate non-convergence. The conventionally acceptable threshold for convergence was $\widehat{R} < 1.2$ (Gelman and Rubin 1992). More recently, Vehtari et al. (2019) proposed a more stringent threshold of $\widehat{R} < 1.01$.

Following the same notation, we define autocorrelation as the correlation between two subsequent θ-values within the same chain (Lynch 2007, ~147). In this study we only consider AC at lag 1, i.e., the correlation between the t^{th} and $t + 1^{th}$ iteration of the same chain.

$$AC = \left(\frac{T}{T-1}\right)\frac{\sum_{t=1}^{T-1}(\theta_t - \bar{\theta}^{(\cdot m)})(\theta_{t+1} - \bar{\theta}^{(\cdot m)})}{\sum_{t=1}^{T}(\theta_t - \bar{\theta}^{(\cdot m)})^2}.$$

We can interpret AC-values as a measure of stationarity. If AC-values are close to zero, there is no dependence between subsequent samples within imputation chains. Positive AC-values, however, indicate recurrence. If θ-values of subsequent iterations are similar, trending may occur. Negative AC-values show no threat to the stationarity component of convergence. On the contrary even—negative AC-values indicate that θ-values of subsequent iterations diverge from one-another, which may increase the variance of θ and speed up convergence. As convergence diagnostic, the interest is therefore in positive AC-values. Moreover, the magnitude of AC-values may be evaluated statistically, but that is outside of this note's scope.

In short, convergence is reached when there is no dependency between subsequent iterations of imputation chains ($AC = 0$), and chains intermingle such that the only difference between the chains is caused by the randomness induced by the algorithm ($\widehat{R} = 1$).

Simulation Hypothesis This study evaluates whether $\widehat{R}$ and AC could diagnose convergence of multiple imputation algorithms. We assess the performance of the two convergence diagnostics against the recommended evaluation criteria for MI methods (i.e., average bias, average confidence interval width, and empirical coverage rate across simulations; Van Buuren 2018, ~2.5.2). That is, there is no baseline measure available to evaluate performance against.The current practice of visually inspecting imputation chains for signs of non-mixing or non-stationarity can only diagnose severely pathological cases of non-convergence. We could also look at distributional characteristics, and plausibility of imputed values, see Vink (n.d.). For now, this is outside of the scope of this study.

Based on an empirical finding (Lacerda, Ardington, and Leibbrandt 2007), we hypothesize that $\widehat{R}$ will over-estimate non-convergence of MI algorithms. The threshold of $\widehat{R} < 1.01$ will then be too stringent for diagnosing convergence. This over-estimation may, however, be diminished because $\widehat{R}$ can falsely diagnose convergence if initial values of the algorithm are not appropriately over-dispersed (Brooks and Gelman 1998, p~437). In `mice`, initial values are chosen randomly from the observed data. Therefore, we cannot be certain that the initial values are over-dispersed. We expect this to have little effect on the hypothesized performance of $\widehat{R}$. No hypothesis was formulated about the performance of AC as convergence diagnostic.

Methods

We use model-based simulation in `R` to perform multiple imputation under 100 simulation conditions. In each condition, the MI algorithm comprises of a different imputation chain length ($T = 1, 2, \ldots, 100$). The number of simulation runs per condition is 1000. For each simulation condition (i.e., number of iterations) and each repetition (i.e., simulation run) we compute convergence diagnostics ($\widehat{R}$ and autocorrelation) and simulation diagnostics (bias, confidence interval width and

coverage). These diagnostics are aggregated across repetitions to obtain their average value per simulation condition. The simulation set-up is summarized in the pseudo-code below. The complete `R` script of the simulation study is available from github.com/gerkovink/shinyMice.

```
# pseudo-code of simulation
simulate data
for (number of simulation runs from 1 to 1000)
  for (number of iterations from 1 to 100)
    create missingness
    impute the missingness
    compute convergence diagnostics
    perform analysis
    pool results
    compute simulation diagnostics
aggregate convergence and simulation diagnostics
```

Data Simulation The point of origin for all simulation conditions is a finite population of $N = 1000$. The data are simulated to solve a multiple linear regression problem, where dependent variable Y is regressed on independent variables X_1, X_2 and X_3:

$$Y \sim \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3.$$

The quantity of scientific interest is regression coefficient β_1. The data generating model is a multivariate normal distribution with the following means structure and variance-covariance matrix:

$$\begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ \epsilon \end{pmatrix} \sim N \left[\begin{pmatrix} 12 \\ 3 \\ 0.5 \\ 0 \end{pmatrix}, \begin{pmatrix} 4 & 4 & 1.8 & 0 \\ 4 & 16 & 4.8 & 0 \\ 1.8 & 4.8 & 9 & 0 \\ 0 & 0 & 0 & 100 \end{pmatrix} \right].$$

Data is simulated with the function `mvtnorm::rmvnorm()`, and outcome variable Y is subsequently calculated as

$$Y = 2X_1 + .5X_2 - X_3 + \epsilon.$$

The complete data is 'amputed' once for each simulation repetition. That is, the function `mice::ampute()` is used to impose a missingness mechanism upon the data. The missingness is univariate, and the probability to be missing is the same

for all four variables, namely 20% (`prop = 0.8, mech = "MCAR"`). This leaves 20% of the rows completely observed. %The resulting amputed data is equal for all simulation conditions in the same repetition.

Missing datapoints are imputed with the function `mice::mice()`. All MI procedures are performed with Bayesian linear regression imputation (`method = "norm"`), and five imputation chains (`m = 5`). The number of iterations varies between simulation conditions (`maxit = 1, 2, \dots, 100`).

From the imputation step, we extract chain means and chain variances across iterations (θ at $t = 1, 2, \ldots, T$).
We compute convergence diagnostics separately for both θs, and for each of the four variables in the dataset. We report the maximum (absolute) value as the 'worst linear predictor' of convergence. $\widehat{R}$ is computed by implementing Vehtari et al.'s (2019) recommendations. AC is computed with function `stats::acf()`.

To estimate the quantity of scientific interest, Q, we perform multiple linear regression on each completed dataset with the function `stats::lm()`. We obtain an estimated regression coefficient per imputation, which are pooled into a single estimate, $\bar{Q}$. We use the function `mice::pool()` to get variance estimates according to Rubin's (1987) rules, and subsequently implement finite population pooling conform Vink and Buuren (2014).

We compute bias as the difference between $\bar{Q}$ and Q. Confidence interval width (CIW) is defined as the difference between the lower and upper bound of the 95% confidence interval (CI95%) around $\bar{Q}$. We compute the CI95% bounds as

$$\bar{Q} \pm t_{(m-1)} \times SE_{\bar{Q}},$$

where $t_{(m-1)}$ is the quantile of a t-distribution with $m - 1$ degrees of freedom, and $SE_{\bar{Q}}$ is the square root of the pooled variance estimate. From bias and CIW, we calculate empirical coverage rates. Coverage rate is the proportion of simulations in which Q is between the bounds of the CI95% around $\bar{Q}$.

Results

Feedback:

- Remove the table.
- Add more info about figure legends and axes.

Figures 2 and 3 display results per simulation condition ($T = 1, 2, \ldots, 100$).

Convergence Diagnostics It is apparent that there is a relation between the number of iterations per simulation condition (T) and the convergence diagnostics. Generally speaking, conditions with longer imputation chains (higher T) coincide with less signs of non-convergence ($\widehat{R}$-values approach one, and AC-values approach

zero). We see more or less equivalent trends for chain means versus chain variances. We, therefore, only discuss the $\widehat{R}$ and AC-values of chain means.

Figure 2A shows that $\widehat{R}$-values generally decrease with increasing imputation chain lengths. The decline stabilizes somewhere between the simulation conditions $T = 30$ and $T = 50$. The downward trend is most pronounced for $T = 3$, and between $T = 5$ and $T = 10$. In the intervening conditions ($3 \leq T \leq 5$), however, we observe a steep increase in $\widehat{R}$-values. This increase implies that non-convergence is under-estimated, and convergence should not be diagnosed at this point. Based on the conventional threshold $\widehat{R} < 1.2$, we would falsely diagnose convergence at $T = 3$. According to the widely used threshold $\widehat{R} < 1.1$, convergence would be diagnosed for conditions where $T > 9$. If we use the recently recommended threshold $\widehat{R} < 1.01$, we would conclude that convergence is reached in none of the simulation conditions.

The AC-values displayed in Figure 2B are almost uniformly increasing as a function of T. The only AC-value that deviates from this observation is for $T = 2$. The lowest AC-value is obtained for $T = 3$. %At $T = 5$, the AC-value reaches the level observed at $T = 2$. The gradual increase plateaus between $T = 10$ and $T = 30$. AC-values in conditions where $T > 70$ are indifferentiable from zero, indicating stationarity. We see an initial decrease between $T = 2$ and $T = 3$. Simulation conditions where $T > 5$ have AC-values greater than $T = 2$.

According to this diagnostic, none of the simulation conditions show signs of non-convergence, since we only observe negative or zero AC-values. The negative AC-values, however, indicate increasing divergence between subsequent samples in conditions where $T < 4$. It may not be wise to terminating the algorithm at that point, but after 'recovering' from this dip in AC-values. This would imply $T = 6$ as the minimal number of iterations.

Taken together, we see that $T > 3$ is the minimal requirement to diagnose convergence ($\widehat{R} < 1.2; AC \leq 0$). This threshold is, however, not sufficient, since we overlook the increase in $\widehat{R}$-values up-to conditions where $T > 5$, and the convergence diagnostics only reach stability at $T > 20$.

Simulation Diagnostics We use average bias, average confidence interval width, and coverage rate as performance measures to evaluate $\widehat{R}$ and AC. We make the general observation that the simulation diagnostics behave as theorized for most simulation conditions (bias around zero, stable confidence interval widths, nominal coverage rate at 95%).

Figure 3A shows that bias is fairly stable across simulation conditions. The condition $T = 1$ clearly deviates from this trend with a negative bias. The bias for $T = 2$ is below average, but within the range of fluctuations. Conditions where $T > 3$ can be diagnosed as unbiased, but the average bias across iterations (as shown by a flat Loess line) only reaches stability at $T = 20$.

CIW is only clearly divergent in the simulation condition $T = 1$, see Figure 3B. However, under-estimating the variance of $\bar{Q}$, which is the case for $T = 3$, may yield spurious inferences. Only conditions where $T > 3$ are therefore considered sufficient. These conditions have similar CIWs, but across iterations stability is not established (i.e., the Loess line is never completely flat within the 100 simulation conditions). %After about twenty-five iterations the average CIW across all simulation conditions and repetitions is reached 0.927.

The empirical coverage rate across repetitions seems more or less stable for conditions where $T > 1$, see Figure 3C. We see some over-coverage at $T = 2$, but that is better than under-estimating the variance of $\bar{Q}$. On average, the coverage rate is somewhat higher than the expected nominal coverage of 95% (Neyman 1934), namely 95%. Similar to the CIWs, there is some trending across iterations (i.e., the Loess line is never flat).

From the simulation diagnostics, we observe that unbiased estimates with nominal coverage rates were obtained in conditions where $T > 3$. This suggests that as little as four iterations may be sufficient for the MI algorithm under the current circumstances.

In short, there is a discrepancy between what the convergence diagnostics and what the performance measures indicate. While $T = 4$ seems sufficient with respect to simulation quantities, the condition where $T = 4$ resulted in the 'one-but-worst' values of both convergence diagnostics. Complete algorithmic convergence as indicated by $\widehat{R}$ and autocorrelation is not reached in conditions where $T < 20$.

Discussion

This note shows that convergence diagnostics $\widehat{R}$ and AC may diagnose convergence of multiple imputation algorithms, but their performance differs from conventional applications to iterative algorithmic procedures.
$\widehat{R}$ and autocorrelation indicate that algorithmic convergence may only be reached after twenty or even forty iterations, while unbiased, confidence valid estimates estimates may be obtained with as little as four iterations. These results are in agreement with the simulation hypothesis: $\widehat{R}$ over-estimates the severity of non-convergence when applied to MI procedures. %This may be due to the quantity of scientific interest chosen. More 'complicated' Qs (e.g., higher order effects or variance components) might show bias, under- or over-coverage at higher T.

According to this simulation study, the recently proposed threshold of $\widehat{R} < 1.01$ may be too stringent for MI algorithms. Under the relatively easy missing data problem of the current study, the threshold was not reached. The other extreme of the $\widehat{R}$-thresholds, the conventionally acceptable $\widehat{R} < 1.2$, may be too lenient for MI procedures. Applying this threshold to the current data, lead to falsely diagnosing convergence at $T = 3$. It appears that the widely used threshold of $\widehat{R} < 1.1$ suits MI algorithms the best. We might, however, also formulate a new

threshold, specifically for the evaluation of MI algorithms. The current study suggests that $\widehat{R} < 1.05$ may be implemented, since that is the level at which the $\widehat{R}$ stabilize (around $T = 20$).

The negative AC-values obtained in this study show no threat of non-stationarity. However, initial dip in AC-values may have implications for the default number of iterations in `mice` (`maxit = 5`). Terminating the algorithm at $T = 5$ may not be the most appropriate, since this lead to the worst convergence, as indicated by $\widehat{R}$ and AC. Under the current specifications, $T > 20$ would be more appropriate.

The observed dip in AC implies that default maxit value of five iterations is the worst possible number of iterations. Moreover, the results of this study imply that assessing the stationarity component of convergence with AC might be redundant, since high AC-values are implausible in MI procedures. That is, the randomness induced by the MI algorithm effectively mitigates the risk of dependency within chains. AC would thus not be informative of the convergence of MI algorithms.

Since this study only considers only an MCAR missingness mechanism, results may not be extrapolated to other missing data problems. Proper performance of the convergence diagnostics under MCAR is necessary but not sufficient to demonstrate appropriateness of $\widehat{R}$ and AC as convergence diagnostics. This is just a proof of concept.

Further research is needed to investigate their performance under clear violation of convergence, e.g. dependency between predictors (predictors with very high correlations). Until then, we have only shown that the convergence diagnostics can diagnose non-convergence of MI algorithms that trend towards a converged state. Also for future research, implement Wernicke diagnostic, and look at developing a convergence diagnostic for substantive models, and implement a Wald test for $AC = 0$.

References

Allison, Paul D. 2001. *Missing Data*. Sage publications.

Brooks, Stephen P., and Andrew Gelman. 1998. "General Methods for Monitoring Convergence of Iterative Simulations." *Journal of Computational and Graphical Statistics* 7 (4): 434–55. https://doi.org/10.1080/10618600.1998.10474787.

Cowles, Mary Kathryn, and Bradley P Carlin. 1996. "Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review." *Journal of the American Statistical Association* 91 (434): 883–904.

Gelman, Andrew, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. 2013. *Bayesian Data Analysis*. Philadelphia, PA, United States: CRC Press LLC.

Gelman, Andrew, and Donald B. Rubin. 1992. "Inference from Iterative Simulation Using Multiple Sequences." *Statistical Science* 7 (4): 457–72. https://doi.org/10.1214/ss/1177011136.

Geweke, John. 1992. "Evaluating the Accuracy of Sampling-Based Approaches to the Calculations of Posterior Moments." *Bayesian Statistics* 4: 641–49.

Hoff, Peter D. 2009. *A First Course in Bayesian Statistical Methods*. Springer Texts in Statistics. New York, NY: Springer New York. https://doi.org/10.1007/978-0-387-92407-6.

Lacerda, Miguel, Cally Ardington, and Murray Leibbrandt. 2007. "Sequential Regression Multiple Imputation for Incomplete Multivariate Data Using Markov Chain Monte Carlo." University of Cape Town, South Africa.

Lynch, Scott M. 2007. *Introduction to Applied Bayesian Statistics and Estimation for Social Scientists*. Springer Science & Business Media.

MacKay, David JC, and David JC Mac Kay. 2003. *Information Theory, Inference and Learning Algorithms*. Cambridge university press.

Murray, Jared S. 2018. "Multiple Imputation: A Review of Practical and Theoretical Findings." *Statistical Science* 33 (2): 142–59. https://doi.org/10.1214/18-STS644.

Neyman, Jerzy. 1934. "On the Two Different Aspects of the Representative Method: The Method of Stratified Sampling and the Method of Purposive Selection." *Journal of the Royal Statistical Society* 97 (4): 558–625. https://doi.org/10.2307/2342192.

Raftery, Adrian E, and Steven Lewis. 1991. "How Many Iterations in the Gibbs Sampler?" Washington University Seatle, Department of Statistics, United States.

R Core Team. 2019. *R: A Language and Environment for Statistical Computing*. Vienna, Austria. https://www.R-project.org/.

Rubin, Donald B. 1987. *Multiple Imputation for Nonresponse in Surveys*. Wiley Series in Probability and Mathematical Statistics Applied Probability and Statistics. New York, NY: Wiley.

Schafer, Joseph L. 1997. *Analysis of Incomplete Multivariate Data*. Chapman; Hall/CRC.

Takahashi, Masayoshi. 2017. "Statistical Inference in Missing Data by MCMC and Non-MCMC Multiple Imputation Algorithms: Assessing the Effects of Between-Imputation Iterations." *Data Science Journal* 16 (July): 37. https://doi.org/10.5334/dsj-2017-037.

Van Buuren, Stef. 2018. *Flexible Imputation of Missing Data*. Chapman; Hall/CRC.

Van Buuren, Stef, and Karin Groothuis-Oudshoorn. 2011. "Mice: Multivariate Imputation by Chained Equations in R." *Journal of Statistical Software* 45 (1): 1–67. https://doi.org/10.18637/jss.v045.i03.

Vehtari, Aki, Andrew Gelman, Daniel Simpson, Bob Carpenter, and Paul-Christian Bürkner. 2019. "Rank-Normalization, Folding, and Localization: An Improved $\widehat{R}$ for Assessing Convergence of MCMC," March. http://arxiv.org/abs/1903.08008.

Vink, Gerko. n.d. "Towards a Standardized Evaluation of Multiple Imputation Routines."

Vink, Gerko, and Stef van Buuren. 2014. "Pooling Multiple Imputations When the Sample Happens to Be the Population." *arXiv:1409.8542 [Math, Stat]*, September. http://arxiv.org/abs/1409.8542.