Multiple Imputation of Missing Values in Economic Surveys: Comparison of Competing Algorithms

Masayoshi Takahashi[1,2] and Takayuki Ito[1]
[1] National Statistics Center, Tokyo, JAPAN
[2] Corresponding author: Masayoshi Takahashi, e-mail: mtakahashi@nstac.go.jp

Abstract

There are many competing computational algorithms in multiple imputation. To this date, however, it is unknown which of these algorithms outperforms the others under what circumstances. In this paper, we describe the mechanisms of various multiple imputation algorithms and compare their performance in a variety of situations to determine which algorithm is best suited to the imputation of missing values in official economic statistics.

Keywords: expectation-maximization with bootstrapping (EMB), fully conditional specification (FCS), joint modeling (JM), Markov chain Monte Carlo (MCMC), multivariate imputation by chained equations (MICE), official economic statistics

1. Introduction

Due to the missing values in a dataset, not only available data size shrinks and efficiency decreases, but also bias is likely to exist if there is a systematic difference between respondents and non-respondents. Therefore, we almost always need to deal with missing values in one way or another, and multiple imputation has been proposed as a method to handle missing data (Rubin, 1987). While the theoretical concept of multiple imputation is simple and has been around for decades, the implementation is difficult and contentious because making a random draw from the posterior distribution is a complicated matter. As a result, there are many competing computational algorithms in software. To this date, however, it is unknown which of these algorithms outperforms the others under what circumstances.

In this paper, we describe the mechanisms of various multiple imputation algorithms and compare their performance in a variety of situations to determine which algorithm is best suited to the imputation of missing values in official economic statistics. Each of the algorithms will be judged in many dimensions, such as accuracy in comparison with the true values, computational efficiency, and so on. A real application on Turnover in the EDINET (Electronic Disclosure for Investors' NETwork) data will be used to illustrate the arguments.

2. Notations and Assumptions of Missing Mechanisms

Let D an $n \times p$ dataset (n = sample size, p = number of variables). If there are no missing values, the distribution of D is normal with mean vector μ and variance-covariance matrix Σ, i.e., $D \sim N_p(\mu, \Sigma)$. Let i refer to observation index, where $i = 1, \ldots, n$. Let j refer to variable index, where $j = 1, \ldots, p$. Let $D = \{Y_1, \ldots, Y_p\}$, where Y_j is the j-th column in D and Y_{-j} is the complement of Y_j, i.e., all columns in D except Y_j. Let R a response indicator matrix. The dimensions of D and R are the same, and whenever D is observed, R = 1; otherwise, R = 0. Also, let Y_{obs} observed data and Y_{mis} missing data: $D = \{Y_{obs}, Y_{mis}\}$.

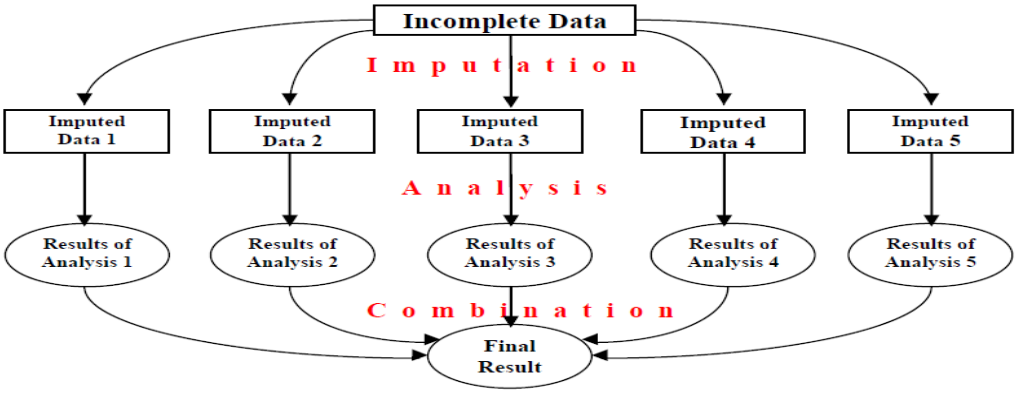
The first assumption is Missing Completely At Random (MCAR), where $P(R|D) = P(R)$. The second assumption is Missing At Random (MAR), where $P(R|D) = P(R|Y_{obs})$. The third assumption is NonIgnorable (NI), where $P(R|D)$ cannot be simplified: R is not independent of D (Little and Rubin, 2002).

3. Multiple Imputation: A Primer

The theory of multiple imputation was first proposed by Rubin (1978). This section briefly summarizes the basic mechanisms of Rubin's multiple imputation (Rubin, 1987; Schafer, 1999; King *et al.*, 2001; Takahashi and Ito, 2012). In multiple imputation, we replace missing values by M simulated values, where $M > 1$. For this purpose, a posterior

distribution of the missing data is constructed, conditional on the observed data. Then, a random draw is made from this posterior distribution, and M multiply-imputed datasets will be created, reflecting the uncertainty about imputation. Separately utilizing each of the M multiply-imputed datasets, we carry out statistical analyses and combine the results of the M statistical analyses to calculate a point estimate. Multiple imputation (M = 5) is schematically shown in Figure 3.1.

Figure 3.1: Schematic Overview of Multiple Imputation



Since we assume a multivariate normal distribution, the imputation model of missing values is linear. Let Y_{ij} be missing. $Y_{i,-j}$ refers to all of the observations, except variable Y_j, in row i. $\tilde{Y}_{ij}$ is an imputed value, which is obtained using equation (1), where ~ signifies random sampling from an appropriate posterior distribution, β is a regression coefficient, and ε stands for fundamental uncertainty.

$$\tilde{Y}_{ij} = Y_{i,-j}\tilde{\beta} + \tilde{\varepsilon}_i \tag{1}$$

Note that the information needed for the calculation of regression coefficients is the mean, variance, and covariance, all of which we can find in μ and Σ. Therefore, if μ and Σ are fully known, the true regression coefficient β can be deterministically calculated based on Y_j, and missing values can be deterministically imputed. In this case, the likelihood function of complete data is equation (2).

$$L(\mu, \Sigma|D) \propto \prod_{i=1}^{n} N(Y_i|\mu, \Sigma) \tag{2}$$

Unfortunately, missing values almost always exist in most datasets. We assume MAR in forming the likelihood of observed data Y_{obs}, i.e., $P(R|D) = P(R|Y_{obs})$. Let us define $Y_{i,obs}$ an observed value of row i in D, $\mu_{i,obs}$ a subvector of μ, and $\Sigma_{i,obs}$ a submatrix of Σ. Since the marginal densities are normal, the likelihood function of observed data Y_{obs} is equation (3).

$$L(\mu, \Sigma|Y_{obs}) \propto \prod_{i=1}^{n} N\left(Y_{i,obs}|\mu_{i,obs}, \Sigma_{i,obs}\right) \tag{3}$$

Since we do not fully know μ and Σ, we cannot know β with certainty. $\tilde{\beta}$ in equation (1), as opposed to $\hat{\beta}$ (ordinary least squares estimate of β), implies this estimation uncertainty. Using the traditional methods, it is not easy to compute equation (3) and to randomly draw μ and Σ from this posterior distribution (Allison, 2002). In order to solve this problem, various computational algorithms have been proposed in the literature, which we will explain in the next section. As of this writing, the relative superiority of these algorithms is not fully known.

4. Competing Algorithms and Software Programs

4.1 Markov Chain Monte Carlo (MCMC): Data Augmentation

The original version of multiple imputation proposed by Rubin is based on the well-known Bayesian computational algorithm, called Markov chain Monte Carlo (MCMC), which is proper imputation (Rubin, 1987; Schafer, 1997).

Monte Carlo is a simulation technique, where a set of independent simulated values are generated based on some probability distribution. A Markov chain is a stochastic process in which the probability of moving from one position to another at time *t* in the series depends only on the current position, θ_t, in the series; thus, it is conditionally independent of the preceding values, $\theta_0, \dots, \theta_{t-1}$. The basic mechanism of MCMC is that, if this chain has run infinitely long, it will find the targeted posterior distribution of interest; therefore, we can generate summary statistics from these values by letting the chain wander around. The surprising trick behind MCMC is that, although each of the simulated values from joint or conditional distributions is serially correlated, these values can be, eventually, regarded as independent draws from the marginal distributions. Data augmentation is an MCMC computational technique, which makes a successive substitution of improved estimates conditional on the preceding value and thus forms a Markov chain. The basic mechanism of data augmentation is that, starting from an initial value θ_0, we generate imputations from the distribution of missing values given the observed values (imputation step) and generate parameter values from the posterior distribution (posterior step), and then repeat these two steps (Schafer, 1997; Little and Rubin, 2002; Gill, 2008):

I-Step: Generate $Y_{mis}^{(t+1)}$ based on $P(Y_{mis}|Y_{obs}, \theta_t)$
P-Step: Generate θ_{t+1} based on $P\left(\theta|Y_{obs}, Y_{mis}^{(t+1)}\right)$
where θ is an unknown parameter and *t* refers to the number of iterations.

In computer software programs, this type of algorithm is made possible by joint modeling (JM), where imputations are drawn from the conditional distribution of the multivariate distribution for the missing data (van Buuren and Groothuis-Oudshoorn, 2011). If the underlying joint distribution can be approximated by the multivariate normal, then the statistical analysis will be guaranteed to be valid (Drechsler, 2009). Examples of the software using this algorithm are R Package Norm 3.0.0 (Schafer, 2008)[1] and SAS PROC MI 9.3 (SAS Institute Inc., 2011).[2]

4.2 Fully Conditional Specification (FCS): Chained Equations

One alternative algorithm to MCMC is Fully Conditional Specification (FCS), in which the method of imputing multivariate missing data is on a variable-by-variable basis. In other words, an imputation model is specified for each incomplete variable, and then imputations are iteratively created for each variable. In this algorithm, the multivariate distribution $P(Y, R|\theta)$ is specified by way of a series of conditional densities $P\left(Y_j|Y_{-j}, R, \lambda_j\right)$, through which Y_j is imputed, given Y_{-j} and R, where λ is the unknown parameters of the imputation model. First, the marginal distribution is used to make a simple random draw. Then, imputation is iterated over the conditionally specified imputation models (van Buuren, 2012). There are many conditionally specified imputation models, but the most prominent is the MICE algorithm, which stands for the Multivariate Imputation by Chained Equations and works as follows.

Based on the observed values in the dataset and the response mechanism, we specify an imputation model for each variable Y_j, which is $P\left(Y_{j,mis}|Y_{j,obs}, Y_{-j}, R\right)$. Then, for each variable *j*, we fill in starting imputations $\tilde{Y}_{j,0}$ by making random draws from the observed values $Y_{j,obs}$. We repeat this process for $t = 1, \dots, T$. We also repeat this process for $j = 1, \dots, p$. At iteration *t*, $\tilde{Y}_{-j,t} = \left(\tilde{Y}_{1,t}, \dots, \tilde{Y}_{j-1,t}, \tilde{Y}_{j+1,t-1}, \dots, \tilde{Y}_{p,t-1}\right)$ is the complete data

[1] Norm 3.0.0 works only in R 2.9.2 or before.
[2] SAS 9.3 experimentally implements an FCS option, but we did not use this experimental feature in SAS.

except Y_j. Draw the unknown parameters of the imputation model, given the observed values, the imputations at *t*, and the response mechanism; in other words, draw $\tilde{\lambda}_{j,t} \sim P(\lambda_{j,t}|Y_{j,obs}, \tilde{Y}_{-j,t}, R)$. Then, draw imputations $\tilde{Y}_{j,t} \sim P(Y_{j,mis}|Y_{j,obs}, \tilde{Y}_{-j,t}, R, \tilde{\lambda}_{j,t})$ (van Buuren, 2012).

One advantage of FCS to JM is that imputation is possible even when there are no suitable multivariate distributions (van Buuren and Groothuis-Oudshoorn, 2011). Examples of the software using this algorithm are R Package MICE 2.13 (van Buuren and Groothuis-Oudshoorn, 2011), PASW Missing Values 18 (SPSS Inc., 2009), and SOLAS 4.01 (Statistical Solutions, 2011).[3]

4.3 Expectation-Maximization with Bootstrapping (EMB)

Another emerging algorithm is the Expectation-Maximization with Bootstrapping (EMB), which combines the traditional methods of the expectation-maximization and the non-parametric bootstrapping.

In the EM algorithm, we first assume a certain distribution and the starting values for the mean and the variance. Using these tentative starting values, an expected value of model likelihood is calculated, the likelihood is maximized, model parameters are estimated that maximize these expected values, and then the distribution is updated. We repeat the expectation and the maximization steps until the values converge, whose properties are known to be a maximum likelihood estimate. Formally, the expectation-maximization can be summarized as follows (Schafer, 1997; Watanabe and Yamaguchi, 2000; Little and Rubin, 2002). Starting from an initial value θ_0, repeat the following two steps:

E-step: $Q(\theta|\theta_t) = \int l(\theta|Y)\, P(Y_{mis}|Y_{obs}; \theta_t) dY_{mis}$, where $l(\theta|Y)$ is log likelihood.
M-step: Maximize $\theta_{t+1} = \arg\max_\theta Q(\theta|\theta_t)$ with respect to θ.
Under certain conditions, it is proven that $\theta_t \to \hat{\theta}$ $(t \to \infty)$.

The non-parametric bootstrapping method utilizes the observed sample as the pseudo-population: A subsample of size *n* is randomly drawn from this observed sample of size *n* with replacement, and this process is repeated *M* times (Wooldridge, 2002).

Combining these two algorithms, the EMB algorithm works as follows. Suppose that, in some incomplete data (sample size = *n*), *q* values are observed and $n - q$ values are missing. We first apply the non-parametric bootstrapping method to obtain bootstrap subsamples of size *n* to be drawn from this incomplete data *M* times. Next, the EM algorithm is applied to each of these *M* bootstrap subsamples to calculate *M* point estimates of μ and Σ, which allows us to impute missing values by forming *M* equations of equation (1). As a result, *M* multiply-imputed datasets are constructed (Congdon, 2006; Honaker and King, 2010). Unlike the above two algorithms, we do not need to resort to the Cholesky decomposition[4] in the bootstrap method and do not need to make a draw from the χ^2 distribution (van Buuren, 2012). Therefore, it is expected to be computationally more efficient than the MCMC methods. R package Amelia II (version 1.6.1) is the software using this algorithm (Honaker, King, and Blackwell, 2011).

5. Results of Comparison Using the EDINET Data & the Simulation Data[5]

The data we used are the EDINET data (Financial Services Agency, 2011). Our dataset has three variables (n = 3,042). Turnover (unit = million yen) is our dependent variable, in which we artificially create missing values. The other two variables are the explanatory variables: Worker (unit = person) and Capital (unit = million yen).

Intuitively, the number of workers implies the size of manpower in a company; thus, as the number of workers increases, the amount of turnover is expected to increase. Also, the amount of capital signifies a company's business size; thus, as the amount of capital

[3] SOLAS is an example of FCS, but does not iterate (van Buuren and Groothuis-Oudshoorn, 2011).
[4] The Cholesky Decomposition (a.k.a the Cholesky factorization) is that if *A* is a positive symmetric definite matrix, i.e., $A = A'$, then there is a matrix *H* such that $A = HH'$, where *H* is lower triangular with positive diagonal elements (Leon, 2006).
[5] Due to limited space, we showed a sketch of the results. Details will be reported during the presentation.

increases, the amount of turnover is expected to increase. The model we used is natural logarithm. In the original dataset, there are no missing values in Turnover, meaning that we know the true values in this variable. For the purpose of experiment, we artificially created missing values in Turnover, making a super variable *x*, which combines information from Worker and Capital. The entire dataset was sorted in the increasing order of *x*. Then, we deleted the values of Turnover when the values of *x* are small. Finally, we deleted *x*. Thus, the missing mechanism is MAR. The rate of missing values is 30%.

Using the EDINET data, we compared the performance of the above-mentioned multiple imputation software programs as follows.[6] First, we set the number of multiply-imputed datasets as 20. We used 100 simulation seeds and compared the 6 software programs; thus, there are a total of 600 trials. We compared the difference between the true mean of Turnover and that of the multiply-imputed Turnover variable. We also checked the difference between the true standard deviation of Turnover and that of Turnover based on the multiply-imputed datasets. Furthermore, we compared the true regression coefficients and those based on the multiply-imputed datasets. Finally, we compared the true *t*-statistics and those based on the multiply-imputed datasets.

Table 5.1 presents preliminary, tentative results (as of April 30, 2013), which shows the average of the absolute differences between the true value and the estimates from each program based on the 100 trials. As for the estimation of the mean of turnover, the differences between SOLAS[7] and SAS/SPSS are statistically insignificant at the 99% level, respectively. The differences between SOLAS and NORM/MICE/AMELIA are statistically significant at the 99% level, respectively. NORM performs better than SOLAS. On the other hand, SOLAS performs better than MICE and AMELIA. However, the substantive differences between SOLAS and these three programs are not large. As for the estimation of the standard deviation of turnover, none of the differences among the programs is statistically significant at the 99% level. Also, as for the estimation of the regression coefficient and its associated *t*-statistics, none of the differences among the programs is statistically significant at the 99% level. Thus, what we basically found is that there are no significant differences across the six software programs in terms of the accuracy of imputation.

Table 5.1: Average of Absolute Differences (Welch Two-Sample *t*-test)

	SOLAS	NORM	SAS	MICE	SPSS	AMELIA
mean	0.007	*0.004	0.006	*0.010	0.005	*0.011
std. dev.	0.048	0.048	0.047	0.049	0.047	0.047
reg. coef.	0.050	0.050	0.049	0.049	0.049	0.049
t-statistic	6.836	6.983	6.787	6.389	6.745	6.302

Note: std. dev. refers to the standard deviation. reg. coef. refers to the regression coefficient of worker on turnover. *t*-statistic refers to the *t*-statistics of worker on turnover. * shows that the difference between each program's estimate and that of SOLAS is statistically significant at the 99% level.

Table 5.2 shows whether each program can handle a simulated gigantic dataset. We found that AMELIA ran quite fast followed by SAS, MICE, SPSS, and SOLAS. We also found that Norm was not even able to run this large dataset. Thus, there are significant differences in terms of the computational efficiency among the software programs.

Table 5.2: Simulation (500,000 observations, 10 variables, MCAR, missing rate = 30%)

	SOLAS	NORM	SAS	MICE	SPSS	AMELIA
time	8m8s	Not Run	4m3s	5m59s	6m30s	3m17s

Note: time = the time to complete multiple imputation ($M = 5$) on the large dataset, m = minutes, s = seconds, and Not Run = the program not able to complete multiple imputation on this large dataset. The maximum number of iterations is set to 20.

[6] This paper is by no means the first of comparing various multiple imputation algorithms. Allison (2000) and Horton and Lipsitz (2001) are probably two of the first research articles. Also see Allison (2002), Horton and Kleinman (2007), and Lin (2010) for the historical development of various multiple imputation algorithms. This paper aims to represent the state of the art of multiple imputation, because most of these algorithms have gone through extensive updates over the past several years.

[7] We thank Statistical Solutions for providing us with a free edition of SOLAS 4.01 for this experiment.

6. Conclusions

This paper described the mechanisms of various multiple imputation algorithms and compared their performance. We showed that none of the multiple imputation algorithms was clearly superior about the accuracy of imputation, but that some algorithms were markedly superior to the others as to the computational efficiency. However, it is too early to make a final call as of this writing, since different algorithms are expected to work differently in various settings. In future research, we intend to diversify the experimental settings to account for this.

References

1. Allison, Paul D. (2000). "Multiple Imputation for Missing Data: A Cautionary Tale," *Sociological Methods and Research* vol.28, no.3: 301-309.
2. Allison, Paul D. (2002). *Missing Data*. CA: Sage Publications.
3. Drechsler, Jörg. (2009). "Far From Normal - Multiple Imputation of Missing Values in a German Establishment Survey," *Work Session on Statistical Data Editing, UNECE*, Neuchâtel, Switzerland, October 5-7, 2009.
4. Financial Services Agency, the Japanese Government. (2011). *EDINET-Electronic Disclosure for Investors' NETwork*. http://info.edinet-fsa.go.jp (Accessed on April 30, 2013).
5. Gill, Jeff. (2008). *Bayesian Methods—A Social Sciences Approach*, Second Edition. London: Chapman & Hall/CRC.
6. Honaker, James and Gary King. (2010). "What to do About Missing Values in Time Series Cross-Section Data," *American Journal of Political Science* vol.54, no.2: 561–581.
7. Honaker, James, Gary King, and Matthew Blackwell. (2011). "Amelia II: A Program for Missing Data," *Journal of Statistical Software* vol.45, no.7.
8. Horton, Nicholas J. and Ken P. Kleinman. (2007). "Much Ado About Nothing: A Comparison of Missing Data Methods and Software to Fit Incomplete Data Regression Models," *The American Statistician* vol.61, no.1: 79-90.
9. Horton, Nicholas J. and Stuart R. Lipsitz. (2001). "Multiple Imputation in Practice: Comparison of Sotfware Packages for Regression Models with Missing Variables," *The American Statistician* vol.55, no.3: 244-254.
10. King, Gary, James Honaker, Anne Joseph, and Kenneth Scheve. (2001). "Analyzing Incomplete Political Science Data: An Alternative Algorithm for Multiple Imputation," *American Political Science Review* vol.95, no.1: 49-69.
11. Leon, Steven J. (2006). *Linear Algebra with Applications*, Seventh Edition. Upper Saddle River, NJ: Pearson/Prentice Hall.
12. Lin, Ting Hsiang. (2010). "A Comparison of Multiple Imputation with EM Algorithm and MCMC Method for Quality of Life Missing Data," *Quality & Quantity* vol.44, no.2: 277-287.
13. Little, Roderick J. A. and Donald B. Rubin. (2002). *Statistical Analysis with Missing Data*, Second Edition. New Jersey: John Wiley & Sons.
14. Rubin, Donald B. (1978). "Multiple Imputations in Sample Surveys - A Phenomenological Bayesian Approach to Nonresponse," *Proceedings of the Survey Research Methods Section, American Statistical Association*: 20–34.
15. Rubin, Donald B. (1987). *Multiple Imputation for Nonresponse in Surveys*. New York: John Wiley & Sons.
16. SAS Institute Inc. (2011). *SAS/STAT 9.3 User's Guide*. Cary, NC: SAS Institute Inc.
17. Schafer, Joseph L. (1997). *Analysis of Incomplete Multivariate Data*. London: Chapman & Hall/CRC.
18. Schafer, Joseph L. (1999). "Multiple Imputation: A Primer," *Statistical Methods in Medical Research* vol.8: 3-15.
19. Schafer, Joseph L. (2008). *NORM: Analysis of Incomplete Multivariate Data under a Normal Model, Version 3*. Software Package for R. University Park, PA: The Methodology Center, the Pennsylvania State University.
20. SPSS Inc. (2009). *PASW Missing Values 18*. Chicago, IL: SPSS Inc.
21. Statistical Solutions. (2011). *SOLAS Version 4.0 Imputation User Manual*. http://www.solasmissingdata.com/wp-content/uploads/2011/05/Solas-4-Manual.pdf. (Accessed on April 30, 2013).
22. Takahashi, Masayoshi and Takayuki Ito. (2012). "Multiple Imputation of Turnover in EDINET Data: Toward the Improvement of Imputation for the Economic Census," *Work Session on Statistical Data Editing, UNECE*, Oslo, Norway, September 24-26, 2012.
23. Watanabe, Michiko, and Kazunori Yamaguchi. (2000). *EM Algorithm to Fukanzen Data no Shomondai (EM Algorithm and the Problems of Incomplete Data)*. Tokyo: Taga Shuppan.
24. van Buuren, Stef and Karin Groothuis-Oudshoorn. (2011). "mice: Multivariate Imputation by Chained Equations in R," *Journal of Statistical Software* vol.45, no.3.
25. van Buuren, Stef. (2012). *Flexible Imputation of Missing Data*. London: Chapman & Hall/CRC.
26. Wooldridge, Jeffrey M. (2002). *Econometric Analysis of Cross Section and Panel Data*. Cambridge, MA: MIT Press.