Multiple Imputation with Mplus

Tihomir Asparouhov and Bengt Muthén
Version 2

September 29, 2010

1 Introduction

Conducting multiple imputation (MI) can sometimes be quite intricate. In this note we provide some general guidance on this process using Mplus. The statistical modeling behind the multiple imputation method in Mplus Version 6 is somewhat complex. The Bayesian estimation method used for the imputations is also quite intricate to use in certain situations. Usually imputations are conducted on large data sets which lead to large models with a large number of parameters. Such large data sets and models can lead to many different complications.

In Section 2 we review the imputation methods available in Mplus. In Section 3 we present some simulated examples and evaluate the performance of different imputation methods. In Section 4 we present some basic tips that could be used to avoid imputation complications and that make the imputation process more transparent and manageable.

2 Multiple Imputations Methods Implemented in Mplus

In Mplus Version 6 multiple imputation (MI) of missing data can be generated from an MCMC simulation. This method was pioneered in Rubin (1987) and Schafer (1997). The imputed data sets can be analyzed in Mplus using any classical estimation methods such a maximum-likelihood and weighted least squares (WLS).

The missing data is imputed after the MCMC sequence has converged. Mplus runs 100 MCMC iterations and then stores the generated missing data values. The process is repeated until the desired number of imputations have been stored. These imputed missing data sets are essentially independent draws from the missing data posterior. The missing data can be imputed in Mplus from a single-level or from a two-level model. The data can be imputed from an unrestricted model ($H1$ model), which we call $H1$ imputation, or it can be imputed from any other model that can be estimated in Mplus with the Bayesian estimator, which we call $H0$ imputation. Unrestricted models are general enough so that model misspecification can not occur. However, these models have a large number of parameters and convergence is often difficult to achieve, particularly for large multivariate sets with many variables that include combinations of categorical and con-

tinuous. Unrestricted two-level models can also have convergence problems because of the large number of parameters estimated on the between level sometimes using only a limited number of two-level units/clusters. In case of convergence problems with the H1 imputations, the H0 imputation offers a viable alternative as long as the estimated model used for the imputation fits the data well. With H0 imputation some ground breaking opportunities arise, such as, imputation from LCA models and factor analysis models.

Three different unrestricted H1 models have been implemented in Mplus for the H1 imputation. All three models are defined for the combination of categorical and continuous variables. Prior to estimating the H1 model all continuous variables are standardized to mean zero and variance one. After estimation the continuous variables are transformed back to their original scale. In the following sections we describe the three H1 imputation models.

2.1   Variance Covariance Model

In this model all variables in the data set are assumed to be dependent variables. For each categorical variable Y_j in the model, taking the values from 1 to k, we assume that there is a underlying continuous latent variable Y_j^* and threshold parameters $\tau_{1j}, ..., \tau_{k-1j}$ such that

$$Y_j = t \Leftrightarrow \tau_{t-1j} \leq Y_j^* < \tau_{tj} \qquad (1)$$

where we assume $\tau_{0j} = -\infty$ and $\tau_{kj} = \infty$. The above definition essentially converts a categorical variable Y_j into an unobserved continuous variable Y_j^*. Let Y be the vector of all observed continuous dependent variables and all underlying continuous latent variables the model is given by

$$Y = \nu + \varepsilon \qquad (2)$$

where ε is a zero mean vector with variance covariance matrix Θ which is one full block of unrestricted variance covariance matrix with 1s on the diagonal for each categorical variable in the model. In addition the vector ν has means fixed to 0 for all categorical variables and free for all continuous variables. For all categorical variables we estimate also all thresholds as defined in (1).

The two-level version of this model is as follows

$$y = \nu + \varepsilon_w + \varepsilon_b \qquad (3)$$

3

where ε_w and ε_b are zero mean vectors defined on the within and the between level respectively with variance covariance matrices Θ_w and Θ_b. Both of these matrices are one full block of unrestricted variance covariance. Again the vector ν has means fixed to 0 for all categorical variables and free for all continuous variables. For all categorical variables we estimate also all thresholds again as defined in (1). If a variable is specified as within-only variable the corresponding component in the ε_b vector is simply assumed to be 0, which implies that also the corresponding parameters in the variance covariance matrix Θ_b are 0. Similarly if a variable is specified as between-only variable the corresponding component in the ε_w vector is simply assumed to be 0, which implies that also the corresponding parameters in the variance covariance matrix Θ_w are 0. For categorical variables for identification purposes again the variance of the variable in Θ_w is fixed to 1, with the exception of the case when the categorical variable is between-only. In that case the variance on the between level in Θ_b is fixed to 1.

This model is the default imputation model in all cases.

2.2   Sequential Regression Model

In this model all variables in the data set are assumed to be dependent variables as well. The model is defined by the following equations

$$y_1 = \nu_1 + \beta_{12}y_2 + \beta_{13}y_3 + ... + \beta_{1p}y_p + \varepsilon_1 \tag{4}$$

$$y_2 = \nu_2 + \beta_{23}y_3 + \beta_{24}y_4 + ... + \beta_{2p}y_p + \varepsilon_2 \tag{5}$$

$$...$$

$$y_p = \nu_p + \varepsilon_p \tag{6}$$

where $\varepsilon_1,...,\varepsilon_p$ are independent residuals with variances $\theta_{11},...,\theta_{pp}$. Essentially in this model we have replaced the parameters θ_{ij}, $i < j$ in the variance covariance model described in the previous section with the regression parameters β_{ij}, $i < j$. For two-level models this θ_{ij} to β_{ij} conversion is basically applied to both levels. The identification restrictions needed for categorical variables are as for the variance covariance model.

The above model was pioneered in Raghunathan et al. (2001). It is particularly powerful and useful in the case of combination of categorical and continuous variables when used also in the framework of observed mediators,

see Asparouhov and Muthén (2010a). Note that depending on how the mediator is treated we actually have two different models for H1 imputation defined here, i.e., sequential regression with observed mediators and sequential regression with latent mediators, see Asparouhov and Muthén (2010a). The default is the observed mediator model. This model is the easier to estimate among the two models.

2.3 Regression Model

In this model all variables in the data set that have missing data are assumed to be dependent variables Y and all variables that do not have missing data are assumed to be covariates X. The model is defined by

$$y = \nu + Kx + \varepsilon \tag{7}$$

where ν and ε are as in the variance covariance model. The two level generalization for this model is also simply a generalization of the two-level variance covariance model with the addition to the covariates. For two-level models each covariate is classified as either within-only or between-only, i.e., each covariate is used on just one of the two levels.

One advantage of this model is that if only a few variables have missing values the unrestricted model will have much fewer number of parameters then the previous two models and will likely reach convergence faster.

3 Examples

3.1 Estimating Structural Equation Models With Categorical Variables and Missing Data

The most popular method for estimating structural equation models with categorical variables is the weighted least squares method (estimator=WLSMV in Mplus). This method however has certain limitations when dealing with missing data. The method is based on sequentially estimating the univariate likelihood and then conditional on the univariate estimates the bivariate model is estimated. The problem with this approach is that when the missing data is MAR and one dependent variable Y_1 affects the missing data mechanism for another variable Y_2, the two variables have to be estimated

simultaneously in all stages of the estimation otherwise the estimates will be biased.

The weighted least squares estimator relies on unbiased estimates of tetrachoric, polychoric and polyserial correlations to build estimates for any structural model. If these correlation estimates are biased the structural parameters estimates will also be biased. Consider for example the growth model of 5 binary variables observed at times $t = 0, 1, 2, 3, 4$. The model is described by the following equation

$$P(Y_{it} = 1) = \Phi(\eta_{1i} + t\eta_{2i}).$$

where Φ is the standard normal distribution function. The model has 5 parameters: the mean μ_1 of the random intercept η_{1i} and the mean μ_2 of the random slope η_{2i} as well as the variance covariance Ψ of these two random effects which has 3 more parameters. We generate 100 data sets of size 1000 and we generate missing data for y_2, y_3, y_4 and y_5 via the following missing data mechanism, for $j = 1, ..., 4$

$$P(Y_j \text{ is missing}|Y_1 = 0) = Exp(-2)/(1 + Exp(-2)) \approx 12\% \qquad (8)$$

$$P(Y_j \text{ is missing}|Y_1 = 1) = Exp(1)/(1 + Exp(1)) \approx 73\%. \qquad (9)$$

Thus y_1 affects the missing data mechanism for $y_2, ..., y_4$. This missing data mechanism is MAR (missing at random).

The results of this simulation study can be found in Table 1. We analyze the data using the true model with several different estimators. We analyze the data with the WLSMV estimator directly. In addition, using the Mplus imputation method we analyze the data with the WLSMV estimator with 5 imputed data sets as well as 50 imputed data sets. The multiple imputation method is based on a Bayesian estimation of an unrestricted model which is then used to impute the missing values. Multiple and independent imputations are created which are then analyzed using Rubin (1987) method. The unrestricted model used for imputation is the unrestricted variance covariance model. The parameter values used in this simulation study are as follows $\mu_1 = 0.00$, $\mu_2 = 0.20$, $\psi_{11} = 0.50$, $\psi_{22} = 0.50$, and $\psi_{12} = 0.30$.

As expected we see that the WLSMV estimates are biased. In particular the mean of the random slope is underestimated dramatically by the WLSMV estimator. Also the coverage for the WLSMV estimator is unacceptable. On the other hand, the WLSMV estimator with 5 imputed data

Table 1: Bias(Coverage) for MAR dichotomous growth model.

Estimator	μ_1	μ_2	ψ_{11}	ψ_{22}	ψ_{12}
WLSMV	-0.03(.92)	-0.16(.02)	-0.23(.62)	0.09(.96)	-0.08(.68)
WLSMV (5 Imput.)	-0.01(.95)	-0.01(.92)	0.07(.90)	0.04(.91)	0.00(.94)
WLSMV (50 Imput.)	-0.01(.94)	-0.01(.92)	0.06(.94)	0.03(.93)	0.00(.95)

sets and the WLSMV estimator with 50 imputed data sets performed very well both in terms of bias and coverage and there doesn't appear to be a substantial difference between these two estimators, i.e., increasing the number of imputed data sets from 5 to 50 does not seem to improve the results. The 5 imputed data sets are sufficient.

3.2 Imputation Example with Large Number of Continuous Variables

In this section we will illustrate some of the issues that can be encountered with a difficult imputation problem and the resolutions available in the Mplus framework. First we generate a data set with 50 variables using a factor analysis model

$$Y_j = \nu_j + \lambda_j \eta + \varepsilon_j$$

where $\nu_j = 0$, $\lambda_j = 1$ and η and ε_j are standard normal variables. We generate a single data set of size 1000. We also generate missing values for this data set according to the missing data mechanism, for $j = 1, ..., 40$

$$P(Y_j \text{ is missing}) = \frac{1}{1 + Exp(\sum_{k=41}^{50} Y_k)}.$$

With the above missing data mechanism we generate missing values for the first 40 variables while the last 10 variables have no missing values. The missing data generation is MAR, rather than MCAR or NMAR, because the last 10 variables influence the probability of missing and those variables are always observed.

We use only this one data set but we conduct imputations with 4 different sample sizes $N = 70, 100, 200$ and 1000 simply by analyzing the first N

observations instead of the entire data set. With every imputation method we generate 5 imputed data sets. As a simple measure of the quality of the imputation we use the following mean squared error

$$MSE_1 = \sqrt{\frac{1}{50}\sum_{j=1}^{50}(\bar{\mu}_j - \hat{\mu}_j)^2} \tag{10}$$

where $\bar{\mu}_j$ is the average mean of Y_j over the five imputed data sets and $\hat{\mu}_j$ is the corresponding ML estimate when the data is analyzed using the true one factor analysis model. A second measure of the quality of the estimates is

$$MSE_2 = \sqrt{\frac{1}{50}\sum_{j=1}^{50}(\bar{\mu}_j - \mu_j)^2} \tag{11}$$

where μ_j is the true value, in this particular case $\mu_j = 0$.

We use 4 different imputation methods, one $H0$ imputation method and 3 $H1$ imputation methods. The first method is an $H0$ imputation method based on a one factor analysis model using the PX parameterization, see Asparouhov and Muthén (2010b). The PX parameterization is needed here because of the large number of variables and small sample sizes. In addition to the $H0$ imputation we conduct 3 types of $H1$ imputations all based on the unrestricted mean and variance covariance model. The difference between the three $H1$ imputations is in the priors used for the variance covariance matrix. The three priors we used are $IW(I, p+1)$, $IW(0,0)$ and $IW(I, -p-1)$, where p is the size of the variance covariance matrix, see Asparouhov. and Muthén (2010b). The first of these priors is the default prior in Mplus while the other two priors are currently not available in Mplus for imputation although formally speaking they could be used through setting up the $H1$ imputation model as an $H0$ imputation model. In small sample sizes the priors on the variance covariance parameters tend to be quite influential with the Bayesian estimation. The results are presented in Table 2 and Table 3.

When we impute the data using sample size $N = 1000$ the convergence of the imputation models is very fast, convergence occurs within 1000 MCMC iterations for the $H1$ imputations for example. When we impute the data using sample size $N = 200$ the convergence of the imputation model is a bit slower, convergence occurs within 10000 MCMC iterations. The convergence with sample sizes $N = 100$ and $N = 70$ is much slower and with two of the

Table 2: MSE_1 for different imputation methods.

Sample Size	$H0$	$H1$ $IW(0, -1-p)$	$H1$ $IW(0,0)$	$H1$ $IW(I, p+1)$
70	0.168	-	-	0.355
100	0.110	-	-	0.289
200	0.054	0.204	0.118	0.111
1000	0.020	0.044	0.039	0.038

Table 3: MSE_2 for different imputation methods and FIML.

Sample Size	FIML	$H0$	$H1$ $IW(0, -1-p)$	$H1$ $IW(0,0)$	$H1$ $IW(I, p+1)$
70	0.437	0.423	-	-	0.562
100	0.357	0.369	-	-	0.530
200	0.143	0.156	0.251	0.179	0.156
1000	0.059	0.065	0.070	0.068	0.068

priors the $H1$ imputation model did not converge at all. In this particular example the $H1$ model is not identified because any one of the first 40 variables has fewer observations than there are variables in the model, see point 6 in Section 4. Thus the $H1$ model in the Bayesian framework for this example is identified primarily by the prior assumption and therefore the prior has a substantial effect on the estimated $H1$ imputation model. This means also that the imputed values are influenced substantially by the priors. The two MSE measures appear to be leading to the same conclusions in all cases. Among the three $H1$ imputations the best appears to be the one based on the $IW(I, p+1)$ prior, which is the Mplus default prior. In general the $H0$ imputations appear to be more accurate than any of the $H1$ imputations. In practical applications however the one factor model used in the $H0$ imputation could be inaccurate and that could lead to additional error in the estimates due to minor or major misspecifications of the $H0$ imputation model.

3.3 Two-level Imputation Example with Large Number of Continuous Variables

In this section we illustrate the imputation methods for two-level continuous data. We generate multivariate data with 50 variables for M clusters, each of size 30, according to the following two-level factor analysis model

$$Y_{jik} = \nu_j + \lambda_{wj}\eta_{wik} + \lambda_{bj}\eta_{bk} + \varepsilon_{bjk} + \varepsilon_{wjik}$$

where Y_{jik} is the $j-$th observed variable, $j = 1, ..., 50$, for observation i, $i = 1, ..., 30$ in cluster k, $k = 1, ..., M$. The latent factor variable η_{wik} is the within level factor variable for observation i in cluster k. The latent factor variable η_{bk} is the between level factor variable for cluster k. The variables ε_{wjik} and ε_{bjk} are the residual variables on the within and the between level for the j-th variable. To generate the data we use the following parameter values. The loading parameters λ_{wj} and λ_{bj} are set to 1. The residual variances and the factor variances are also set to 1. The intercept parameter ν_j is set to 0. We conduct five different imputation methods, three $H1$ imputation methods, using as in the previous section the three different priors for the variance covariance matrix: $IW(I, -p - 1)$, $IW(0, 0)$ and $IW(I, p + 1)$. The $H1$ imputation is based on the unrestricted mean and variance covariance two-level model. We also include two $H0$ imputation methods, both based on a two level factor analysis model with one factor on the within level and one factor on the between level, i.e., the $H0$ imputation model is the same as the model used to generate the data. The two $H0$ imputations differ in the parameterization of the imputation model. For the first $H0$ imputation we use as in the previous section the PX parameterization. For the second $H0$ imputation we use the L parameterization, see for details Asparouhov and Muthén (2010b). When the number of clusters is small the PX parameterization yields much better results than the L parameterization, but it is not clear if this advantage will materialize into an imputation advantage. In the two-level estimation the total sample size is large, i.e., the within level sample size is large. When the sample size is large the choice of the parameterization is irrelevant and therefore for simplicity we choose the L parameterization for the within level factor model. Thus the first $H0$ imputation is really based on a PX parameterization on the between level and an L parameterization on the within level. The second $H0$ imputation is based on the L parameterization on both levels. Using each of the imputation methods we generate

10

Table 4: MSE_1 for the interprets in two-level models.

Number of Clusters	$H0$ PX	$H0$ L	$H1$ $IW(0,-1-p)$	$H1$ $IW(0,0)$	$H1$ $IW(I,p+1)$
40	0.041	0.049	-	-	0.066
80	0.023	0.027	-	-	0.036
200	0.012	0.015	0.019	0.022	0.020

5 imputed data sets which are than analyzed using the maximum-likelihood estimator for the true two-level factor analysis model.

To evaluate the performance of the different imputation methods we compute (10) for both the 50 intercept parameters and the 50 between level loadings. The intercept parameters are typically influenced the most by missing data treatment. In addition the between level loadings are affected by the choice of the parameterization in the two-level factor model and thus we will compute (10) also for the 50 between level loadings.

Tables (4) and (5) contain the results. It is clear here again that the $H0$ imputation is more accurate than the $H1$ imputation. This is again because the $H0$ imputation is correctly specified. Among the two $H0$ imputations the more accurate is the PX parameterization particularly when the number of clusters is 40 and 80. Among the $H1$ imputations we encountered again convergence problems with the priors $IW(0,-1-p)$ and $IW(0,0)$ when the number of clusters is 40 or 80. When the number of clusters is 200 the differences between the 3 methods is small as expected since the prior assumptions have little effect on the estimates when the sample size is large. Overall it appears that the among the three $H1$ imputations the one based on the $IW(I,1+p)$ is the best choice. The $IW(I,1+p)$ prior is the default prior in Mplus for the H1 imputation method.

The above example shows that two-level imputations are quite similar to the single level imputations with one exceptions. While in single level imputations difficulties can arise when the sample size is close to the number of variables, in two-level imputations we see these difficulties when the number of clusters is close to the number of variables. These are precisely the situations when the model is not identified or poorly identified, see point 6 in Section 4.

Table 5: MSE_1 for the between level loadings in two-level models.

Number of Clusters	$H0$ PX	$H0$ L	$H1$ $IW(0, -1-p)$	$H1$ $IW(0,0)$	$H1$ $IW(I, p+1)$
40	0.064	0.089	-	-	0.110
80	0.029	0.042	-	-	0.052
200	0.017	0.017	0.027	0.028	0.025

3.4 Imputation Example with Large Number of Categorical Variables

In this section we illustrate the imputation method for categorical variables. We generate data with 30 categorical variables with sample size 1000 using the following factor analysis model

$$Y_j^* = \lambda_j \eta + \varepsilon_j \tag{12}$$

where

$$Y_j = t \Leftrightarrow \tau_{t-1j} \leq Y_j^* < \tau_{tj}. \tag{13}$$

Each of the categorical variables takes 4 values: $t = 1, ..., 4$. The parameters used to generate the data are $\tau_{0j} = -1$, $\tau_{1j} = 0$, $\tau_{2j} = 1$, and $\lambda_j = 1$. The factor variance and the residual variances in this model are fixed to 1. The variables Y_j for $j = 26, ..., 30$ have no missing data while the variables Y_j for $j = 1, ..., 25$ have missing data generated according to the model

$$P(Y_j \text{ is missing}) = \frac{1}{1 + Exp(-1.5 + 0.1 \sum_{k=26}^{30} Y_k)}.$$

This model produces approximately 30% missing data for each of the Y_j variables $j = 1, ..., 25$. We use a single data set of size 1000 but will conduct imputation for sample size $N = 50, 100, 200, 500,$ and 1000 by simply analyzing the first N observations from the total sample. Five imputed data sets are produced and then analyzed with the WLSMV estimator using the factor analysis model given in (12) and (13). We impute the data using four different imputation methods. The first imputation method is an $H1$ imputation method using the Mplus variance covariance imputation model

12

for categorical data. The second imputation method is an $H1$ imputation method using the Mplus sequential regression imputation model for categorical data with observed covariates. The third imputation method is the $H0$ imputation method based on a one factor analysis model using the PX parameterization. The forth imputation method is the $H0$ imputation method based on a one factor analysis model using the L parameterization. In addition we analyze the unimputed data set with the WLSMV estimator directly using the true model given in (12) and (13). The WLSMV estimator does not support MAR missing data as the one generated here and therefore it is expected to be biased. Note here that the ML estimation method can also be used for the estimation of this data directly. However the ML estimation method would heavily rely on the fact that the data is generated from a one factor analysis model. In general the true model could have many more factors and residual correlations. So in that respect the ML method would not be an appropriate substitute for the imputation methods, because the ML estimation method will be computationally very intensive and does not support residual correlations.

The results of this simulation are presented in Table 6. As a measure of fit here we use MSE_2 given in (11) for all threshold parameters. In all cases the imputation methods outperform the direct unimputed WLSMV method. For sample size $N = 50, 100, 200$ the $H1$ imputation method based on the sequential regression model did not converge so we do not report any results in that case. From this example we can conclude that the $H1$ sequential regression imputation is sensitive to sample size and for small sample sizes this method will likely fail. The $H1$ imputation method based on the variance covariance model and the $H0$ imputation methods converge in all cases. The differences between these imputation methods appears to be small. Note here that the $H0$ imputation method is in general sensitive to correctly specifying the one factor analysis model. However, that sensitivity is much smaller than the ML sensitivity because only the imputed data relies on that assumption. Thus the observed data which usually will dominate the estimation in practical applications will reduce this sensitivity and if there are more factors in the final model they will likely realize in the final WLSMV analysis because of the observed data.

From the results of this simulation study we can make the following conclusions. The $H1$ imputation method based on the variance covariance model appears to be preferable than the $H0$ imputation methods as it is less dependent on correct imputation model specification and the advantages of the

Table 6: MSE_2 for the threshold parameters for single level imputation with categorical variables.

Sample Size	Direct WLSMV	H1-Cov Imputed WLSMV	H1-Seq Imputed WLSMV	H0-PX Imputed WLSMV	H0-L Imputed WLSMV
50	0.311	0.279	-	0.264	0.294
100	0.252	0.189	-	0.192	0.179
200	0.162	0.146	-	0.139	0.145
500	0.151	0.107	0.107	0.104	0.107
1000	0.130	0.071	0.067	0.068	0.069

$H0$ imputation method even when the model is correctly specified appear to be small. Among the two $H1$ imputation methods the variance covariance method appears to be preferable. Among the two $H0$ imputation methods it appears that the PX parameterization has a slight advantage. This advantage appears to be much smaller however than the corresponding advantage for continuous variables.

3.5 Two-level Imputation Example with Large Number of Categorical Variables

In this section we illustrate the multiple imputation methods for two-level data with categorical variables. We generate multivariate data with 30 categorical variables for M clusters, each of size 30, according to the following two-level factor analysis model

$$Y_{jik}^* = \lambda_{wj}\eta_{wik} + \lambda_{bj}\eta_{bk} + \varepsilon_{bjk} + \varepsilon_{wjik} \tag{14}$$

where Y_{jik}^* is the $j-$th underlying normal latent variable, $j = 1, ..., 50$, for observation i, $i = 1, ..., 30$ in cluster k, $k = 1, ..., M$, which is related to the observed categorical variable through the equation

$$Y_{jik} = t \Leftrightarrow \tau_{t-1j} \leq Y_{jik}^* < \tau_{tj}. \tag{15}$$

Each of the categorical variables takes 4 values: $t = 1, ..., 4$. The latent factor variable η_{wik} is the within level factor variable for observation i in cluster k.

The latent factor variable η_{bk} is the between level factor variable for cluster k. The variables ε_{wjik} and ε_{bjk} are the residual variables on the within and the between level for the j-th variable. To generate the data we use the following parameter values. The loading parameters λ_{wj} and λ_{bj} are set to 1. The factor variances are also set to 1. The residual variances on the within level is set to 1 and on the between level it is set to 0.5. The threshold parameters are as in the previous section $\tau_{0j} = -1$, $\tau_{1j} = 0$, and $\tau_{2j} = 1$. The variables Y_j for $j = 26, ..., 30$ have no missing data while the variables Y_j for $j = 1, ..., 25$ have missing data generated according to the model

$$P(Y_j \text{ is missing}) = \frac{1}{1 + Exp(-1.5 + 0.15 \sum_{k=26}^{30} Y_k)}.$$

We use a single data set of size 6000, i.e., a data set with 200 clusters, but we conduct imputation for sample size $N = 1500$ and $N = 6000$, i.e., with $M = 50$ and $M = 200$ clusters. The smaller data set is based again on the first N observations from the entire sample. In addition, we vary the number of variables used in the imputation model. Imputations are conducted with all 30 variables but also a smaller imputation is conducted with 10 variables, using variable $Y_{21}, ..., Y_{30}$. Five imputed data sets are produced in each case and then analyzed with the WLSMV estimator using a two-level factor analysis model given in (14) and (15). We impute the data using three different imputation methods. The first imputation method is an $H1$ imputation method using the Mplus default imputation model for categorical twolevel data which is the unrestricted mean and variance covariance model. The second imputation method is the $H0$ imputation method based on a two-level factor analysis model with one factor on both levels using the PX parameterization on the between level and the L parameterization on the within level. The third imputation method is the $H0$ imputation method based on a two-level factor analysis model with one factor on both levels using the L parameterization on both levels. In addition we analyze the unimputed data set with the WLSMV estimator directly using the correct model given in (14) and (15). The WLSMV estimator does not support MAR missing data for two-level model as well and therefore it is expected to be biased. The ML estimation method can not be used for the estimation of this data using the model given in (14) and (15) because it will require 31 dimensions of integration when $P = 30$ or 11 dimensions of integration when $P = 10$.

The results of this simulation are presented in Table 7. As in the previous section we use as a measure of fit the MSE_2 given in (11) for all threshold

Table 7: MSE_2 for the threshold parameters for two-level imputation with categorical variables.

Number of Clusters	Number of Variables	Direct WLSMV	H1 Imputed WLSMV	H0-PX Imputed WLSMV	H0-L Imputed WLSMV
50	10	0.352	0.268	0.273	0.268
200	10	0.204	0.077	0.078	0.079
50	30	0.418	0.273	0.276	0.278
200	30	0.235	0.074	0.073	0.074

parameters. In all cases the imputation methods performed better than the direct/unimputed WLSMV estimator. The $H0$ and $H1$ imputation methods converge in all cases. The difference in the precision of the $H0$ imputation methods and the $H1$ imputation method appears to be very small. The difference in the precision of the two $H0$ imputation methods also appears to be very small regardless of the fact that the loadings on the between level are inflated in the L parameterization. This is probably due to the fact that this misestimation is filtered out by the categorization of the data.

From the results of this simulation study we can make the following conclusions. The $H1$ imputation method appears to work well. The $H0$ imputation method is also a viable alternative to the $H1$ imputation method.

4 General Tips and Observations

1. The data should always be analyzed first with some basic method. One such tool is available in Mplus. Using the TYPE=BASIC option of the ANALYSIS command is the simplest tool to use. If some of the variables are categorical you can treat them as continuous as a first step. As a second step you can treat them as categorical. TYPE=BASIC is somewhat more advanced when there are categorical variables. If the imputation is a two-level imputation you can first conduct TYPE=BASIC and then as a second step conduct TYPE=BASIC TWOLEVEL. All these analysis will yield some basic information about the data that can be used to diagnose or avoid problems.

2. Using the results of TYPE=BASIC you should make sure that there are no variables that are perfectly correlated, i.e., that there are no variables that have correlation 1 or near 1 (or -1 or near -1). If there are two such variables remove one of the two variables since that additional variable does not carry any additional information. If the correlation is say 0.99 you should probably still remove one of the two variables as it can cause poor mixing and slow convergence. Of course if the imputation works well you do not need to remove such variables from the data set even if they have only a marginal contribution. Polychoric or tetrachoric correlations which are ± 1 may not result in an imputation problem in general. If such correlations do cause a problem then the joint distribution of the two variables has empty cells. In that case we recommend that the two variable be added/subtracted to produce a single variable as a substitute for the two variables. In addition binary indicators with small positive frequency can be combined to improve the mixing.

One common situation when perfectly correlated variables are present in the data is for example when AGE is a variable in a longitudinal study and is recorded for every round of observations. Often the longitudinal observations are collected one year apart and thus AGE2= AGE1+1, which leads to perfectly correlated variables. Only one of the AGE variables should be retained.

Another situation that frequently occurs in practice is with dummy variables. If each possible category of a variable has a dummy indicator then the dummy variables sum to 1, i.e., these variables are linearly dependent and the general imputation model would again be unidentified.

Another common situation in longitudinal studies is when a variable of interest is recorded in every period and in addition the total value from all periods is recorded as well. If all of these variables are included in an imputation model the imputation model will be unidentified. The total value variable carries no additional information and should not be included in the imputation.

3. Do not use all of the variables given on the NAMES list in the VARIABLE command for the multiple imputations. Instead, choose a smaller subset of variables that could be predictive of missingness by using the

USEVARIABLES list of the VARIABLE command. Other variables on the NAMES list that should be saved in the multiple imputation data sets should be put on the AUXILIARY list of the VARIABLE command.

4. Do not use for imputation variables that you know have no predictive power for the missing values. For example, individual ID numbers used in the computer system to identify a person, social security numbers, driver license numbers, group level ID variables, and even zip codes do not have a predictive power and should be removed from the imputation data set because they can cause problems due to among other things having a large scale. Zip codes in principle can contain useful information however they should not be used in raw format as a continuous scale variable because increase in your zip code would not be a meaningful quantity. Instead the zip code variable should be converted to multiple dummy variables / zip code indicators.

5. Unlike other software packages Mplus will impute missing data only after successfully estimating a general/unrestricted one or two-level model with the Bayes estimation method. This means that a certain convergence criterion has to be satisfied before the imputations are generated. One of the problems that can occur in an Mplus imputation is slow mixing and non-convergence. Non-convergence can for example be caused by model non-identification. Below we describe specific common situations that lead to poor mixing, non-identifications and non-convergence. These problems are typically resolved by removing one or several variables from the imputation process using the option USEVAR of the VARIABLE command.

It is possible in Mplus to simply run a fixed number of MCMC iterations and then impute the missing data and essentially ignoring the convergence criteria. This can be done with the FBITER option of the ANALYSIS command. In certain cases such an approach is warranted. For example the imputation model does not need to be identified to produce valid imputations. As an example consider the situation where two variables in the data set are perfectly correlated. A regression model where these two variables are covariates is not identified but the model still can be used for imputations. Using the FBITER option however should be used only as a last resort.

6. The imputation model has to be identified otherwise the estimation will not converge. A basic identifiability requirement for the imputation model is that for each variable in the imputation the number of observations should be at least as many as the number of variables in the imputation model. Suppose that there are 50 variables in the imputation model and that there are 1000 observations in the data set. Suppose that the first variable has 960 missing values and only 40 observed values. Since $40 < 50$ the imputation model is not identified. That variable should be removed from the imputation or it should be imputed from a smaller data sets, for example with 30 variables. If for example there are 60 observed values and 940 missing values the model will be identified but essentially 51 parameters(1 mean parameter, 1 residual variance parameter and 49 regression coefficient parameters) in the imputation model are identified with only 60 observations. This may work, but would likely lead to slow convergence and poor mixing. So even though the model is identified, this variable would cause slow convergence and if the variable is not important one should consider dropping that variable from the imputation.

In two-level imputation models this constraint becomes even more dramatic because the number of variables has to be less than the number of clusters, i.e., if you have 50 variables but only 40 clusters you might have to remove at least half the variables to be able to estimate the imputation model. If the number of variables on the between level is more than the number of clusters then the model is unidentified. If the number of variables is less than the number of clusters but close to that number the model will be formally identified but will likely converge very slowly. More specifically the identifiability requirement for the two-level imputation model is that for each variable the number of clusters with an observed value should be more than the number of variables in the imputation, i.e., variables with large number of missing values will still cause problems. There are several possible alternative resolutions. One is to drop variables from the imputation until the number of variables is less than the number of clusters. Another is to specify within-between variables as within only variables. This way the variables on the between level will be reduced without completely removing variables. This of course should be done for the variables with the smallest ICC, which could be computed ahead of time using

TYPE=BASIC TWOLEVEL. Yet another alternative is to completely switch from an H1 imputation, i.e., imputation from a general two-level model, to H0 imputation. This amounts to for example specifying a two-level model in Mplus, estimating it with the Bayes estimator and using the DATA IMPUTATION command to specify the file names for the imputed data sets. Common H0 imputation models would be for example a one factor analysis model on the between level paired with an unrestricted variance covariance model on the within level or a one factor analysis model on the within level.

7. Some data sets contain missing variable indicators, i.e., for an observed variable with missing values a binary indicator is created which is 1 when the variable is missing and zero otherwise. Those variables will generally cause poor convergence and mixing during the imputation and should be removed. The model essentially is again unidentified. To be more precise in the Baysian framework every model is identified simply because of the priors (as long as the priors are proper). For imputation purposes however models identified only by the prior should be avoided. When a missing variable indicator is present in the data set the correlation between the missing variable indicator and the observed variable is unidentified. Missing variable indicators are created in general to pursue NMAR modeling. Adding the missing variable indicator to the imputation essentially creates a NMAR imputation, which is an advanced topic and probably should be done using H0 imputations with specific and known NMAR models rather than the general and unrestricted imputation models.

Another variable similar to the missing variable indicators in the case of longitudinal studies is the time of drop out variable which is basically the total number of observed values or the time of the last observation. Typically that variable also leads to non-identification because the correlation between the last variable and the drop out variable is unidentified (for observations where the last variable is observed the drop out variable is constant).

8. Another possible problematic outcome with the imputation estimation in Mplus is a situation when the MCMC iterations can not performed at all. There is no slow convergence but rather the iterations are not done at all. For example this can happen if there are perfectly corre-

lated variables in the data. In this case, in the Gibbs sampler, a singular posterior variance covariance matrix is obtained for the parameters and Mplus can not generate samples from the posterior distribution of the parameters. If Mplus does not indicate which variable or variables caused the problem one can simply guess this using a trial and error method. For example if the data set contains 50 variables and imputation can not be obtained using all 50 variables, the data set can be split in two parts with 25 variables each. If one of the two sets can be imputed and the other can not then we know which set of 25 variables contains the problematic variable(s). We can sequentially add variable to the successful imputation run to get the largest imputation data set possible.

9. An example of what will not help with convergence problems in an imputation run is the IMPUTE option of the DATA IMPUTATION command. This option has no effect on the estimation of the imputation model. The option is used only as a data manipulation command. If a variable that has missing values is not on the IMPUTE option list then in the imputed data sets the missing values for that variable are simply not replaced by the MCMC generated values, but are stored as missing values again. For example, that variable can be analyzed with the FIML estimation. This is done for the situations when imputation is desired only for some variables but not for others. The imputation model or its estimation however is not changed by that option. Having all the variables on the IMPUTE list does not decrease or affect the chances of convergence.

10. In terms of convergence the easiest imputation model to estimate is the single level model where all the variables are continuous. When categorical variables are added to the model the estimation becomes somewhat more intricate. In addition extending the imputation to two-level imputation will make the estimation more difficult.

The more variables there are in the model the slower the convergence. The more categorical variables there are in the model the slower the convergence, particularly for two-level models. Two-level data sets are more difficult to impute than single level data sets. If a two-level imputation does not work, try first the single level imputation. If that does not converge try to figure this problem first. In certain situations

it would be beneficial to switch to the REGRESSION or SEQUENTIAL models for imputation purposes, using the MODEL option of the DATA IMPUTATION command. For the REGRESSION models Mplus imputes from a conditional multivariate regression model, where all variables without missing values are conditioned on. If only a few variables have missing values and many variables have all values present, this alternative imputation model can be quite easy to estimate. The SEQUENTIAL model consists of a collection of simple univariate regression equations and it can be quite easy to estimate as well. In case when categorical variable cause slow convergence / poor mixing an approximate alternative would be to estimate the imputation model assuming that the variables are continuous and then use the VALUES option of the DATA IMPUTATION command to essentially round off the imputed values to their categorical levels.

11. Imputation of large data sets is based on the estimation of a large unrestricted model with many parameters. This estimation can be slow simply because there are many parameters in the unrestricted model. The main solution for this problem is to switch to some more restricted H0 imputation using factor analysis models or latent class models. In general if an $H1$ imputation is attempted and it fails due to non-convergence it is possible to use simple factor analysis models for $H0$ imputation. While the $H0$ imputations relies to some extent on the correct specification of the imputation model that specification has only a limited impact on the final data analysis and it appears that minor misspecifications of the imputation model are harmless.

12. In general H1 model imputation works somewhat like a black box. You do not have control of say starting values, priors, or specifically monitoring convergence for individual parameters and other diagnostic tools that are available through the Bayes estimation. If you want such control or if you want to diagnose a non-convergence or any other problem you can always switch from an H1 imputation to an equivalent H0 imputation. You simply have to write the general imputation model in your input file under the MODEL command and request the Bayes estimator. You should always verify that the model is specified correctly by comparing the number of estimated parameters in the H0 and H1 imputations. The number of parameters in the H1 imputation

can be found in the black screen, while the number of parameters in the H0 imputation can be found in the Mplus general output or with the TECH1 option of the OUTPUT command. In addition you should be clear on what type of imputation you are performing, i.e., H0 or H1. Here is a brief summary about the two types of imputation. If the estimator is set to Bayes then you are performing an H0 imputation. If the estimator is not set to Bayes then you are performing an H1 imputation. When you are performing an H0 imputation the data is imputed from the model that is specified in the input file, so you should be sure that the model is correct. If you are conducting an H1 imputation the data is imputed from the general unrestricted model Mplus estimates behind the scenes. The data is imputed from that general model and has nothing to do with the model specified in the input file (if any is specified). There are two types of H1 imputation. First with type=basic you don't need a model specification. The data is simply imputed and stored. Second you have an H1 imputation that includes a specified model. This model is not used for imputation purposes. The imputed data is generated from the unrestricted model Mplus will estimate behind the scenes. The model in the input file is simply the model that is estimated with the already imputed data sets. The Mplus user's guide, see Muthén and Muthén (1998-2010), contains examples of H0 and H1 imputations. Example 11.5 is an H1 imputation while example 11.6 is an H0 imputation.

13. Analysis of imputed data is equivalent to FIML (full information maximum likelihood) analysis of the raw unimputed data. If there is no specific reason to use the imputation method perhaps it should not be used. One reason to use multiple imputations is that the FIML estimation is not available or is much more complicated or much more computationally intensive. Another reason is this. When the data set contains a large number of variables but we are interested in modeling a small subsets of variables it is useful to impute the data from the entire set and then use the imputed data sets with the subset of variables. This way we don't have to worry about excluding an important missing data predictor.

14. If a large data set has to be imputed the computer resources such as memory could be exhausted. An imputation with 150 variables for

example will use an imputation model with more than 10000 parameters. Mplus stores all generated parameter values in the MCMC chain. The options FBITER and THIN in the ANALYSIS command can be used to strictly control the amount of memory Mplus uses. The option FBITER specifies the total number of MCMC iterations that should be generated. The option THIN specifies the number of MCMC iterations to be generated before the generated parameter set is recorded. This option is generally designed to make the generated parameters sets more independent of each other. One possible way to deal with memory problems is to use a fixed FBITER value, and to increase the number of MCMC iterations simply by increasing the THIN option until convergence.

References

[1] Asparouhov, T. and Muthén, B. (2010a) Bayesian Analysis Using Mplus. Mplus Technical Report. http:\\www.statmodel.com

[2] Asparouhov, T. and Muthén, B. (2010b) Bayesian Analysis of Latent Variable Models using Mplus. Mplus Technical Report. http:\\www.statmodel.com

[3] Muthén, L.K. and Muthén, B.O. (1998-2010). Mplus Users Guide. Sixth Edition. Los Angeles, CA: Muthén & Muthén

[4] Raghunathan, T. E., Lepkowski, J. M., Van Hoewyk, J., and Solenberger, P. (2001) A Multivariate Technique for Multiply Imputing Missing Values Using a Sequence of Regression Models. Survey Methodology, Vol. 27, No 1. 85-95.

[5] Rubin, D.B. (1987) Multiple Imputation for Nonresponse in Surveys. J. Wiley & Sons, New York.

[6] Schafer, J.L. (1997) Analysis of Incomplete Multivariate Data. Chapman & Hall, London.