

PHUSE EU Connect 2025, Hamburg, 17NOV2025

Multiple Imputations - overview of missing data problem, solution methods and SAS implementation

Dmytro Kalinin, Principal Statistical Programmer, CSL Vifor, Glattbrugg, Switzerland

Introduction to Missing Data Problem

- **Definition and Causes** - Missing data occurs when information intended for collection is absent due to various reasons like dropouts or equipment failure.
- **Impact on Analysis** - Missing data reduces statistical power and can introduce bias, risking the accuracy of research conclusions.

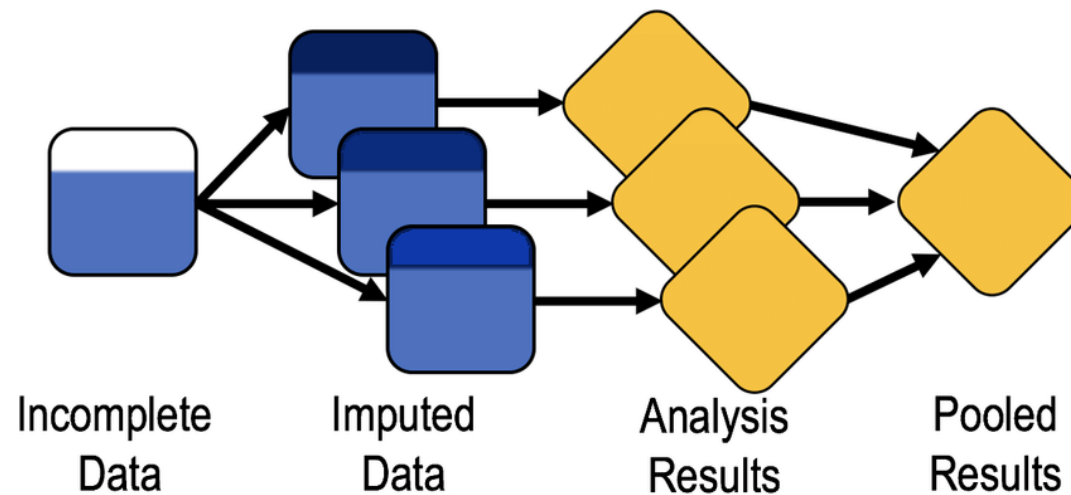
Missing Data Mechanisms

- **Missing Completely at Random (MCAR)** occurs when the missingness is entirely random and unrelated to any data, i.e. the probability that a particular patient will have a missing value does not depend on the patient.
- **Missing at Random (MAR)** happens when the probability of data being missing depends on other observed variables, i.e. the probability that a particular patient will have a missing value may depend on the observed factors (treatment group, baseline characteristic etc).
- **Missing Not at Random (MNAR)** is the most complex case, where the missing data depends on the missing values themselves or unobserved variables, i.e. the probability that a particular patient will have a missing value may depend on unobservable factors, such as the missing value itself.

Distinguishing MAR and MNAR is difficult, complicating the choice of appropriate imputation and analysis methods.

Three Steps of Multiple Imputation

- **Imputation:** Multiple complete datasets are generated by replacing missing values with plausible values.
- **Analysis:** Each of the multiple datasets is analyzed using a standard statistical method.
- **Pooling:** The results (e.g., parameter estimates and their standard errors) from each analysis are combined into a single overall result using Rubin's rules, to account for the uncertainty introduced by the imputation process.



Imputation Step – Key Questions

- **Which variables are intended for imputation?** - Study endpoints are typically the primary targets for imputation, but it is advisable to impute the underlying variables when the endpoint is derived rather than directly measured.
- **Which factors should be included in the imputation model?** - General Recommendation - inclusion of a broader set of relevant factors generally enhances the quality and robustness of the imputation model.
- **How many imputations should be performed?** - Decide on the number of imputed datasets to generate, balancing statistical efficiency and computational feasibility. Use the Relative Efficiency (RE) parameter, which is calculated by PROC MI, RE value greater than 0.95 is considered acceptable.
- **What imputation method and model type should be selected?**

Imputation Step – Missing Data Patterns

- **Monotone** Missing Data Patterns

USUBJID	SEX	AGE	BASE	AVALV1	AVALV2
001	M	47	120	122	124
002	F	32	110	116	.
003	M	25	118	.	.
004	F	27	.	.	.

- **Arbitrary** (Non-Monotone) Missing Data

USUBJID	SEX	AGE	BASE	AVALV1	AVALV2
001	M	47	120	122	124
002	F	32	110	.	116
003	M	25	.	.	120
004	F	.	121	125	120

Imputation Step – Models and Variable Types

Variety of model types:

- **Reg**: Linear regression.
- **Regpmm** (regression predictive mean matching): After regression, the imputed value is selected from among several observed values close to the regression predicted value.
- **Logistic**: Logistic regression.
- **Discrim**: Discriminant function.
- Formally, the Monotone method also has a **Propensity** type, but it is not recommended.

Classification of data types:

- **Continuous** - height, weight, blood sugar level, blood pressure (Reg or Regpmm).
- **Ordinal** - AE severity: mild, moderate, severe; drug doses: low dose, middle dose, high dose (Logistic).
- **Nominal** - Sex, Race, Body locations (Discrim).
- **Binary** - yes/no, success/failure, response/non-response (Logistic).

Imputation Step – Methods

- **MCMC** (Monte-Carlo Markov Chain)
- **Monotone**
- **FCS** (Fully Conditioned Specifications)
- **Hybrid** MCMC/Monotone method

Method	Missing pattern	Variable types	Recommendation
MCMC	Arbitrary	Only continuous	No
Monotone	Monotone	Any type	If monotone pattern
FCS	Arbitrary	Any type	If arbitrary pattern
MCMC/Monotone	Arbitrary	Any type	Use on demand

Proc MI in SAS - Statements

```
proc mi ...;  
  var var1 var2 var3 var4 ...;  
  class var1 var2 ...;  
  mcmc ...;  
  monotone ...;  
  fcs ...;  
run;
```

Main Statements:

- **VAR** - specifies all relevant variables, including those intended for imputation as well as all factors used in the modeling process.
- **CLASS** - The CLASS statement lists those variables that are classificatory (categorical) - as in all SAS procedures.
- **MCMC** <options> - specifies the details of the MCMC method for imputation. Some of the options: CHAIN= single or multiple, we can leave at by default; IMPUTE=Monotone - only until a monotone pattern is achieved, used in the hybrid MCMC/Monotone method.
- **MONOTONE** <model type> (imputed variable = factors. All variables are imputed in turn. Factors will be all variables from the VAR specified before this one, so it is usually not necessary to specify factors after the "=" sign.
- **FCS** <model type> (imputed variable = factors). FCS has the same syntax as MONOTONE. The "CLASSEFFECTS" option is needed if there are classification factors among the factors.

Proc MI in SAS - Options

Main Options:

- **DATA=** specifies the input dataset.
- **OUT=** designates the output dataset.
- **NIMPUTE=** defines the number of imputations to be performed. A simple way to find out the missing data pattern - run proc MI with NIMPUTE=0.
- **SEED=** Sets an arbitrary numeric value to initialize the random number generator. This is essential for quality control (QC), ensuring reproducibility of results across repeated runs.
- **MIDISPLAYPATTERN=** Displays missing data patterns table.
- **MINIMUM=**, **MAXIMUM=**, **ROUND=**. Options are meaningful for MCMC, Monotone/FCS Reg methods. In other cases, the imputed value is always one of the observed values. For example: SEX, AGE, and BASE variables remain unchanged. AVALV1 and AVALV2 are rounded to the nearest integer and are limited to a range of 1 to 100.

```
proc mi minimum=. . . 1 1 maximum=. . . 100 100 round=. . . 1 1;  
  var sex age base avalv1 avalv2;
```

Proc MI Examples

- Method Monotone with Regpmm model

```
proc mi data=indata out=outdata nimpute=50 seed=45780;  
  var weight base chg4 chg8 chg12;  
  monotone regpmm(chg4 chg8 chg12);  
run;
```

- Method FCS with Reg model for two variables

```
proc mi data=indata out=outdata nimpute=50 seed=122001;  
  var trtp skinclass base chg12 chg24;  
  class trtp skinclass;  
  fcs reg(chg12=trtp skinclass base trtp*base chg24);  
  fcs reg(chg24=trtp skinclass base trtp*base chg12);  
run;
```

- Hybrid MCMC/Monotone method with IMPUTE=Monotone option

```
proc mi data=indata out=mono nimpute=50 round=1 seed=45779;  
  by trtpn;  
  var weight base chg4 chg8 chg12;  
  mcmc impute=monotone;  
run;  
  
proc mi data=mono out=outdata nimpute=1 <options>;  
  by _imputation_;  
  var weight base chg4 chg8 chg12;  
  monotone reg (chg4 chg8 chg12);  
run;
```

Analysis Step

The analysis step in multiple imputation involves conducting the same complete-data statistical analysis on each of the separately imputed datasets. Each of the imputed datasets is analyzed separately using any method that would have been chosen had the data been complete. This step can be implemented using any analytical procedure in SAS, e.g., PROC MIXED, PROC GLM, PROC LOGISTIC, PROC NPAR1WAY, PROC FREQ, etc. We apply procedures with the BY _IMPUTATION_ operator and save the results in datasets using the Output Delivery System (ODS) in SAS.

A critical component of this analysis step - addressed in detail through examples in the subsequent "Pooling" chapter - is the specification of a properly structured output dataset. This dataset must contain the estimated statistics and their corresponding standard errors from each repetition of the analysis. It serves as the essential input for the PROC MIANALYZE procedure, which is utilized during the "Pooling" step

Pooling Step - Proc MIANALYZE in SAS

The pooling step in multiple imputation combines parameter estimates and their standard errors from analyses of multiple imputed datasets into a single, combined result using Rubin's rules.

Pooling step is implemented in SAS using **Proc MIANALYZE**. The syntax used for it is contingent upon the analytical approach employed during the preceding analysis step

```
proc mianalyze data=indata;  
  modeleffects est1 est2 ...;  
  stderr StdErr1 StdErr2 ...;  
run;
```

Proc MIANALYZE Examples

- **Processing PROC MIXED, GLM and GENMOD results**

PROC MIXED implements a mixed model, but also ANOVA and ANCOVA. We typically process results from ODS datasets LSMeans and Diffs. Within the PARMs statement, specify the dataset containing the LSMeans or Diffs results.

In the MODELEFFECTS statement, include the same effects as specified in the PROC MIXED procedure for LSMeans - this may involve an interaction term such as trtpn*avisitn.

The CLASS statement should list all classification variables used in the model.

When processing Diffs, the BY trtpn_trtpn statement may be required if multiple comparisons are present.

```
proc mianalyze parms=LSMeans;  
  class trtpn;  
  modeleffects trtpn;  
run;
```

The results in the ODS dataset ParameterEstimates:

Parameter	trtpn	Estimate	Std Error	95% Confidence Limits		Pr > t
trtpn	1	0.771134	0.148230	0.480577	1.061690	<.0001
trtpn	2	1.914720	0.150902	1.618771	2.210669	<.0001

Parameter	trtpn	Estimate	Std Error	95% Confidence Limits		Pr > t
trtpn	1	-1.143586	0.211537	-1.55833	-0.72884	<.0001

Proc MIANALYZE Examples

- Processing Proc LOGISTIC and PHREG results

```
proc logistic data=indata;  
  by _imputation_;  
  class trtpn site / param=ref;  
  model avalc = trtpn site base;  
run;
```

Saving the ODS ParameterEstimates dataset

```
proc mianalyze parms(classvar=classval)=ParameterEstimates;  
  class trtpn site;  
  modeleffects trtpn site;  
run;
```

Differ from PROC MIXED results processing by the CLASSVAR=CLASSVAL option. As a result, we obtain estimates of the model parameters, but we need to obtain the odds ratios.

Parameter	trtpn	site	Estimate	Std Error	95% Confidence Limits		Pr > t
trtpn	1		-0.341673	0.420978	-1.16706	0.483716	0.4171
site		1	0.127304	0.622060	-1.09233	1.346943	0.8379
site		2	-0.693500	0.718109	-2.10162	0.714616	0.3343
site		3	-0.214927	0.646589	-1.48343	1.053577	0.7396
site		4	-0.126253	0.717925	-1.53424	1.281733	0.8604

We need to exponentiate (using the exp function) the parameter estimate and confidence limits.
A similar approach works for the Hazard Ratio in Proc **PHREG**.

Proc MIANALYZE Examples

- **Processing PROC NPAR1WAY results**

```
proc npar1way data=indata wilcoxon;  
  by _imputation_;  
  class trtpn;  
  var aval;  
run;
```

The WilcoxonTest dataset, generated via ODS, contains the variable Z, which represents a test statistic assumed to follow a standard normal distribution.

To prepare this dataset for analysis using PROC MIANALYZE, a DATA step is performed to append a new variable, StdErr, with a constant value of 1 across all records. This modified dataset is then used as input for PROC MIANALYZE to compute the corresponding p-value.

```
proc mianalyze data=WilcoxonTest;  
  modeleffects Z;  
  stderr StdErr;  
run;
```

Proc MIANALYZE Examples

- **Processing PROC FREQ results**

```
proc freq data=indata;  
  by trtpn _imputation_  
  tables avalc / binomial;  
run;
```

Consider a scenario where the objective is to analyze the proportion of patients who achieved a specific response. Although proportions (or percentages) inherently follow a binomial distribution, when the sample size is sufficiently large, the distribution of the estimates approximates normality. Under such conditions, it is appropriate to apply PROC MIANALYZE for further analysis.

The Binomial dataset generated via ODS contains the proportion of interest along with its Asymptotic Standard Error (ASE). To prepare this dataset for analysis using PROC MIANALYZE, it must be transformed - either through a PROC TRANSPOSE step or a DATA step - so that each row corresponds to a unique combination of TRTPN and _IMPUTATION_, with the proportion (_BIN_ variable) and ASE (E_BIN variable) presented as paired variables on a single line.

```
proc mianalyze data=Binomial_transformed;  
  by trtpn;  
  modeleffects _BIN_  
  stderr E_BIN;  
run;
```

Proc MIANALYZE Other Cases

- **Processing Other Cases**

The following types of analysis are difficult to process because the statistics are not normally distributed: Chi-square test, CMH test, Fisher's test, Kaplan-Meier estimates, Log-rank test, Correlation coefficients and many others.

General approach: begin by applying a transformation that normalizes the distribution of the target statistic. Specific formulas for these transformations vary depending on the type of analysis and are documented in specialized statistical literature. Once the data has been transformed, proceed with the analysis using PROC MIANALYZE. Finally, apply the inverse transformation to return the results to their original scale for interpretation

Multiple Imputation and ADaM Datasets

- **Restructure the Dataset Horizontally.** If the imputation model utilizes data from multiple visits or endpoints, the source dataset must first be transposed into a horizontal format to accommodate the model requirements.
- **Execute the Imputation Procedure.** Apply the PROC MI procedure to perform the imputation. It is essential to specify a fixed SEED value to ensure reproducibility and facilitate the quality control (QC) process.
- **Revert to Vertical Structure.** Once the imputation is complete, transpose the dataset back to its original vertical structure for consistency with standard ADaM formats.
- **Identify Imputed Values.** As PROC MI does not inherently distinguish between observed and imputed values, merge the imputed dataset with the original source data using key variables. This allows for the identification of imputed records.
- **Assigning Imputation Data Type.** For each imputed value, assign a DTYPE such as “MI” to clearly indicate its origin.
- **Rename the Imputation Index Variable.** Replace the default _IMPUTATION_ variable name with a valid and meaningful label, such as IMPNO, to align with dataset naming conventions.

References

1. “Multiple Imputations Seminar”. Leonid Zeitlin, Kharkiv, 2022.
2. “Multiple Imputation of Missing Data Using SAS”. Patricia Berglund, Steven G. Heeringa.
3. “Combining Analysis Results from Multiply Imputed Categorical Data”. Bohdana Ratitch, Ilya Lipkovich, Michael O’Kelly: <https://www.lexjansen.com/pharmasug/2013/SP/PharmaSUG-2013-SP03.pdf>
4. Combining Survival Analysis Results after Multiple Imputation of Censored Event Times. Jonathan L. Moscovici, Bohdana Ratitch: <https://www.pharmasug.org/proceedings/2017/SP/PharmaSUG-2017-SP05.pdf>
5. SAS documentation: PROC MI and PROC MIANALYZE.

Q&A