Paper AS04

Multiple imputation as a valid way of dealing with missing data

Vadym Kalinichenko, Intego Group, LLC, Kharkiv, Ukraine

ABSTRACT

Missing data appears in every study. In terms of clinical trials it could be a potential source of bias. Missing data in clinical trials may emerge due to various reasons, e.g. some patients could be prematurely discontinued from the study or could miss planned visits while remaining in the study. Every reasonable effort should be made to obtain the protocol-required data for all the study assessments that are scheduled for all the enrolled patients.

Multiple imputation provides a useful and effective way for dealing with missing data. This process results in valid statistical inferences that properly reflect the uncertainty due to missing values.

This paper reviews methods for analyzing missing data, including basic approach and applications of multiple imputation techniques. It presents SAS (PROC MI and PROC MIANALYZE) and R (MICE package) procedures for creating multiple imputations for incomplete multivariate data, analyzes and compares results from multiple imputed data sets.

INTRODUCTION

The missing information make the data corrupted, introduces an element of bias, invalidates the results and conclusions, makes it unsuitable to apply statistics and makes it liable for rejection by authorities due to the deviations. However, if we simply exclude these patients with missing data it will affect the power of the study. At the same time, it is likely that the patients with missing values are the ones with extreme values (treatment failure, toxicity, and good responders). Exclusion of these patients will lead to underestimation of variability and hence will narrow the confidence interval. Missing data presents various problems. Firstly, the absence of data reduces statistical power, which refers to the probability that the test will reject the null hypothesis when it is false. Second, the lost data can cause bias in the estimation of parameters. Third, it can reduce the representativeness of the samples. Fourth, it may complicate the analysis of the study. Each of these distortions may threaten the validity of the trials and can lead to invalid conclusions. Even though the issues around the missing data are well-documented, it is common practice to ignore missing data and apply analytical techniques which simply delete all the cases having missing data on any of the variables used in the analysis.

MISSING DATA AND MULTIPLE IMPUTATION

It is important to understand how SAS procedures handle missing data. Most SAS statistical procedures exclude observations with any missing variable values from the analysis. It is called listwise deletion and these observations are called incomplete cases. Whereas the usage of only complete cases has its simplicity, it might lead to losing information in incomplete cases. This approach also ignores the possible systematic difference between the complete cases and incomplete cases, and the resulting inference may not be applicable to the population of all cases, especially with a smaller number of complete cases. Some SAS procedures use all the available cases in an analysis, that is, the cases with available information. For example, in PROC MIXED missing values are deleted listwise, i.e., observations with missing values on any of the variables in the analysis are omitted from the analysis. PROC CORR also estimates a correlation by using all the cases with nonmissing values for this pair of variables. This may make better use of the available data, but the resulting correlation matrix may not be definite. Another strategy is single imputation, in which values are substituted at each missing case. Standard statistical procedures for the complete data analysis can then be used with the filled-in data set. For example, each missing value can be imputed from the variable mean of the complete cases. This approach treats missing values as if they were known in the complete-data analyses. Single imputation does not reflect the uncertainty about the predictions of the unknown missing values, and the resulting estimated variances of the parameter estimates will be biased towards zero.

A set of multiple datasets is generated in terms of multiple imputation. The multiply imputed data sets are then analyzed by using standard procedures for complete data and combining the results from these analyses. No matter which complete-data analysis is used, the process of combining of the results from different data sets is essentially the same. Multiple imputation does not attempt to estimate each missing value through simulated values but rather to represent a random sample of the missing values. This process results in valid statistical inferences that properly reflect the uncertainty due to missing values; for example, valid confidence intervals for parameters.

Multiple imputation inference involves three distinct phases:

- The missing data are filled in m times to generate m complete data sets.
- The m complete data sets are analyzed by using standard procedures.
- The results from the m complete data sets are combined for the inference.

Remarkably, m, the number of sufficient imputations, can be only 5 to 10 imputations, although it depends on the percentage of data that are missing. The result is unbiased parameter estimates and a full sample size when done well. Doing multiple imputation well, however, is not always quick or easy. First, it requires that the missing data be ignorable. Second, it requires a very good imputation model. Creating a good imputation model requires knowing your data very well and having variables that will predict missing values.
The MI procedure in the SAS/STAT Software is a multiple imputation procedure that creates multiply imputed data sets for incomplete p-dimensional multivariate data. It uses methods that incorporate appropriate variability across the m imputations. Once the m complete data sets are analyzed by using standard procedures, the MIANALYZE procedure can be used to generate valid statistical inferences about these parameters by combining results from the m complete data sets.

PROC MI

PROC MI provides various methods to create multiply imputed data sets for incomplete multivariate data that can be analyzed using standard SAS procedures. Table 1 summarizes the available statements in PROC MI. The imputation method of choice depends on the pattern of missingness in the data and the type of the imputed variable. For a data set with a monotone missing pattern, the MONOTONE statement can be used to specify applicable monotone imputation methods; otherwise, the MCMC statement can be used assuming multivariate normality.
Multiple Imputation Using SAS Software Statement Description:

BY	Specifies groups in which separate sets of multiple imputations are performed
CLASS	Lists the classification variables in the VAR statement. Classification variables can be either character or numeric.
EM	Uses the EM algorithm to compute the maximum likelihood estimate (MLE) of the data with missing values, assuming a multivariate normal distribution for the data.
FREQ	Specifies the variable that represents the frequency of occurrence for other values in the observation.
MONOTONE	Specifies monotone methods to impute continuous and classification variables for a data set with a monotone missing pattern.
MCMC	Uses a Markov chain Monte Carlo method to impute values for a data set with an arbitrary missing pattern, assuming a multivariate normal distribution for the data.
TRANSFORM	Specifies the variables to be transformed before the imputation process; the imputed values of these transformed variables are reverse-transformed to the original forms before the imputation.
VAR	Lists the numeric variables to be analyzed. If you omit the VAR statement, all numeric variables not listed in other statements are used.

Table 1 PROC MI Statements

The PROC MI statement is the only required statement for the MI procedure.

MISSING DATA MECHANISMS

Before discussing methods for handling missing data, it is important to review the types of missingness. Commonly, these are classified as missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR). An analysis of missing data patterns across contributing participants or centers, over time, or between key treatment groups should be performed to establish the mechanisms behind the missing data.
Missing Completely at Random, MCAR, means there is no relationship between the missingness of the data and any values, observed or missing. Those missing data points are a random subset of the data. There is nothing systematic going on that makes some data more likely to be missing than others. For example, some participants may have missing laboratory values because a batch of lab samples was processed improperly. In these instances, the missing data reduce the analyzable population of the study and consequently, the statistical power.
Missing at Random, MAR, means there is a systematic relationship between the propensity of missing values and the observed data, but not the missing data. In other words, given the observed data, the missingness mechanism does not depend on the unobserved data. For example, a registry examining depression may encounter data that are MAR if male participants are less likely to complete a survey about depression severity than female participants. That is, if probability of completion of the survey is related to their sex (which is fully observed) but not the severity of their depression, then the data may be regarded as MAR.

Missing Not at Random, MNAR, means there is a relationship between the propensity of a value to be missing and its values. This is a case where the people are failed to fill in a depression survey because of their level of depression or the sickest people are most likely to drop out of the study.
MNAR is called “non-ignorable” because the missing data mechanism itself has to be modelled while dealing with the missing data. A model for the missing data mechanism should be specified - that is, the model of how missingness depends on both observed and unobserved quantities.
“Missing Completely at Random” and “Missing at Random” are both considered ‘ignorable’ because we don’t have to include any information regarding the missing data itself while dealing with the missing data.
Why is it important to know which missing data mechanism is presented?
Multiple imputation and Maximum Likelihood assume the data are at least missing at random. So the important distinction here is whether the data are MAR as opposed to MNAR.
Listwise deletion, however, requires the data are MCAR in order not to introduce bias in the results.
As long as the distribution and percentage of missing data are not so great that it negatively affects power, the listwise deletion can be a good choice for MCAR missing data. So the important distinction here is whether the data are MCAR as opposed to MAR.
Keep in mind that in most data sets, more than one variable will have missing data, and they may not all have the same mechanism. It’s worthwhile to diagnose the mechanism for each variable with missing data before choosing an approach.

MISSING DATA PATTERNS

We can distinguish between two main patterns of missingness:

1. Monotone
2. Non-monotone

The definition of monotonic missing is that, once the subject has dropped out he will drop out forever, while for non-monotonic missing the subject may come back or be missing again.
For example, if we follow one subject for five years and he dropped out in the third year, monotonic missing would look like **o o x x x**, and one kind of non-monotonic missing can be **o o x o x**, where

- **o** indicates observed
- **x** indicates missing.

So the third **x** in the non-monotonic missing is like an island. This is just to classify the pattern of missing, and generally the monotonic missing is easier to handle.

Y1	Y2	Y3	Y4
X	X	X	O
X	X	O	O
X	O	O	O

Table 2 Monotone missing pattern

Assessments/variables could be arranged in chronological order or we might even say: missing once - missing forever.

Y1	Y2	Y3	Y4
X	X	X	X
X	X	O	O
X	O	X	O

Table 3 Non-monotone missing pattern

Crucial difference in MI for non-monotone vs. monotone data:

- **non-monotone:** MI requires all the data to be imputed together treating these data as a multivariate response. Same distributional assumptions should be used for both categorical and continuous data.

- **monotone:** MI can approximate the interdependence of the imputations while using one-variable-at-a-time imputation, and MI can thus use different distributional assumptions for continuous and categorical data, as it would normally be done.

WHEN SHOULD MULTIPLE IMPUTATION BE USED TO HANDLE MISSING DATA?

Analysis of observed data (complete case analysis) with the ignorance of the missing data is a valid solution in the following three circumstances.

- The complete case analysis may be used as the primary analysis if the proportions of missing data are below approximately 5% (as a rule of thumb) and it is implausible that certain patient groups (for example, the very sick or the very 'well' participants) specifically are lost to follow-up in one of the compared groups. In other words, if the potential impact of the missing data is negligible, then the missing data may be ignored in the analysis. Best-worst and worst-best case sensitivity analyses may be used if in doubt: first a 'best-worst-case' scenario dataset is generated where it is assumed that all the participants lost to follow-up in one group (referred to as group 1) have had a beneficial outcome (for example, had no serious adverse event); and all those with missing outcomes in the other group (group 2) have had a harmful outcome (for example, have had a serious adverse event). Then a 'worst-best-case' scenario dataset is generated where it is assumed that all the participants lost to follow-up in group 1 have had a harmful outcome; and that all those lost to follow-up in group 2 have had a beneficial outcome . If continuous outcomes are used, then a 'beneficial outcome' might be the group mean plus 2 standard deviations (or 1 standard deviation) of the group mean, and a 'harmful outcome' might be the group mean minus 2 standard deviations (or 1 standard deviation) of the group mean. For dichotomized data, these best-worst and worst-best case sensitivity analyses will then show the range of uncertainty due to the missing data, and if this range does not give qualitatively contradicting results, then the missing data may be ignored. For continuous data imputation with 2 SD will represent a possible range of uncertainty given 95% of the observed data (if normally distributed).
- If only the dependent variable has missing values and auxiliary variables (variables not included in the regression analysis, but correlated with a variable with missing values and/or related to its missingness) are not identified, the complete case analysis may be used as the primary analysis and no specific methods should be used to handle the missing data. No additional information will be obtained by, for example, using multiple imputation but standard errors may increase due to the uncertainty introduced by the multiple imputation.
- As mentioned above, it would also be valid just to perform the complete case analysis if it is relatively certain that the data are MCAR. It is relatively rare that it is certain that the data are MCAR. It is possible to test the hypothesis that the data are MCAR with Little's test, but it may be unwise to build on tests that turned out to be insignificant. Hence, if there is reasonable doubt that the data are MCAR, even if Little's test is insignificant (fail to reject the null hypothesis that data is MCAR), then MCAR should not be assumed.

No assumption	MCAR	MAR	MNAR
	Missing Completely at Random	Missing at Random – ignorability assumption	Missing Not at Random
	Missingness does not depend on the data	Missingness depends only on the observed data	Missingness depends on both observed and missing data
1. LOCF (last observation carried forward) 2.BOCF (baseline value carried forward) 3. WOCF (worst observation carried forward) 4. Imputation based on logical rules	1. CC (Complete-case Analysis) - listwise deletion 2. Pairwise Deletion 3. Available Case analysis 4. Single-value Imputation (for example, mean replacement, regression prediction (conditional mean imputation), regression prediction plus error (stochastic regression imputation) 5. under MCAR, throwing out cases with missing data does not bias your inferences. However, there are many drawbacks	1. Maximum Likelihood using the EM algorithm – FIML (full information maximum likelihood) 2. MMRM (mixed model repeated measurement) – REML (restricted maximum likelihood) 3. Multiple Imputation 4. Two assumptions: the joint distribution of the data is multivariate normal and the missing data mechanism is ignorable 5. Under MAR, it is acceptable to exclude the missing cases, as long as the regression controlled for all the variables that affect the probability of missingness	1. PMM (Pattern-mixture modeling) 2. Jump to Reference 3. Last Mean Carried Forward. 4. Copy Differences in Reference 5. Copy Reference 6. Tipping Point Approach 7. Selection model (Heckman)

Table 4 Missing data assumptions and corresponding imputation methods

MULTIPLE IMPUTATION IMPLEMENTATION

The main steps of the implementation of Multiple Imputation are described below.

Step 1: The analysis starts with observed, incomplete data. Multiple imputation creates several complete versions of the data by replacing the missing values with plausible data values. These plausible values are drawn from a distribution specifically modelled for each missing entry. These imputed datasets are identical for the observed data entries, but differ in the imputed values. The magnitude of these differences reflects our uncertainty about what value to impute.

Step 2: The second step is to estimate the parameters of interest from each imputed dataset. This is typically done by applying the analytic method that we would have used for complete data. The results will differ because their input data differ. It is important to realize that these differences are caused only due to the uncertainty about what value to impute.

Step 3: The last step is to pool the m parameter estimates into one estimate, and to estimate its variance. The variance combines the conventional sampling variance (within-imputation variance) and the extra variance caused by the missing data (between-imputation variance). Under appropriate conditions, the pooled estimates are unbiased and have the correct statistical properties.

SAS IMPLEMENTATION

Random dummy data were generated for the purposes of analysis. Let's assume that the data based on the subjects' responses to a sleep questionnaire(1 to 10) filled in daily. Values at each time point represent averages of daily responses over a period between the visits.

Primary endpoint: change from baseline in sleep quality (daily values on a scale from 1 to 10, averaged) at Week 8.

Primary analysis: ANCOVA with the baseline as a covariate (the most commonly used covariate).

To use PROC MI, the data need to be in a horizontal format as indicated in the example below. Suppose the formatted dataset is called INDS_TR. Variables VIS1-VIS6 correspond to sleep quality at Baseline, Week 1, Week 2, Week 4, Week 6 and Week 8.

SUBJID	TRTN	TRT	Baseline	Week 1	Week 2	Week 4	Week 6	Week 8
1001	1	Treatment A	xx	xx	xx	xx	o	o
1002	2	Treatment B	xx	xx	xx	xx	xx	xx
1003	1	Treatment A	xx	xx	xx	o	xx	xx
1004	2	Treatment B	xx	xx	xx	xx	xx	o
1005	1	Treatment A	xx	xx	xx	xx	o	xx
1006	2	Treatment B	xx	xx	xx	o	o	o

Table 5 Input dataset structure

During the analysis we will go through the following stages:

1. Impute values using PROC MI, then compute change from baseline.
2. Perform ANCOVA to obtain (in each imputed dataset):
   - P-value for the overall treatment effect at Week 8;
   - LSM estimates for the change from baseline in sleep quality for each treatment group at Week 8;
   - LSM estimates for the difference in the change from baseline in sleep quality between Treatment A and Treatment B
3. Combine ANCOVA results from the multiple imputed datasets using PROC MIANALYZE.

The first thing which should be done in terms of multiple imputation is examination of the missing patterns. To do so PROC MI with option NIMPUTE=0 should be used. In this case PROC MI produces no imputation, but the output describes the missing data patterns.

```
/*examine missing patterns*/
proc mi data = inds_tr nimpute = 0;
    var vis1 - vis6;
run;
```

Missing Data Patterns														
									Group Means					
Group	vis1	vis2	vis3	vis4	vis5	vis6	Freq	Percent	vis1	vis2	vis3	vis4	vis5	vis6
1	X	X	X	X	X	X	348	69.60	5.4339	5.4683	5.5260	5.4550	5.6712	5.6233
2	X	X	X	X	X	.	19	3.80	5.1526	6.614	5.1794	4.6033	5.8673	.
3	X	X	X	X	.	X	21	4.20	5.8809	4.7800	4.7276	5.6033	.	6.1085
4	X	X	X	X	.	.	3	0.60	6.4000	5.9700	6.7100	5.1100	.	.

Table 6 Missing Data Patterns

Since output from PROC MI above indicates that the missing pattern is non-monotone, it is necessary to perform next step, which is Partial imputation (just enough to get the monotone missing pattern).
This step should be performed since The MI and MIANALYZE procedures assume that the missing data are missing at random (MAR).

```
/*partial imputation to get monotone missing pattern*/
proc mi data = inds_tr seed=523871 out = data_mono;
    var vis1 - vis6;
    mcmc impute = monotone chain = multiple;
    by trtn;
run;
```

The above procedure will output DATA_MONO dataset with a monotone missing data pattern.
Output from MCMC method contains 5 imputed datasets and becomes an input for imputation of remaining data with regression.
Now let us directly move on to performing multiple imputation.

```
/*perform multiple imputation*/
proc mi data = data_mono out = mono_imp_reg nimpute = 1;
    by _Imputation_;
    var trtn vis1 - vis6;
    class trtn;
    monotone regression;
run;
```

Because input(DATA_MONO) dataset already contains 5 partially imputed datasets, use BY _IMPUTATION_ statement and request only 1 imputation within each BY group.

```
/*ANCOVA for each imputed dataset*/
proc mixed data = mono_imp_reg;
    class trtn;
    model chg_6 = trtn vis1 / solution;
    lsmeans trtn / diff = control('1') cl;
    ods output diffs = lsdiffs lsmeans = lsm solutionf = parms;
    by _Imputation_; /*perform analysis in each imputed dataset*/
run;
```

Request **solution** – parameter estimates for effects with their standard errors (will be needed to get p-value for the treatment effect).
Then use PROC MIANALYZE to combine estimates.

```
proc mianalyze parms(classvar = full) = parms;
    class trtn;
    modeleffects Intercept trtn vis1;
    ods output ParameterEstimates = combined_parms;
run;
```

```
proc mianalyze parms(classvar = full) = lsdiffs;
    class trtn;
    modeleffects trtn;
    ods output ParameterEstimates = combined_lsdiffs;
run;

proc mianalyze parms(classvar=full) = lsm;
    class trtn;
    modeleffects trtn;
    ods output ParameterEstimates = combined_lsm;
run;
```

R IMPLEMENTATION

R can successfully impute missing values as well as SAS. 5 most commonly used R packages for missing value imputation are:

1. MICE
2. Amelia
3. missForest
4. Hmisc
5. mi

In this article MICE package will be used for illustration purposes.

MICE (Multivariate Imputation via Chained Equations) is one of the commonly used packages by R users. Creating multiple imputations as compared to a single imputation (such as mean) takes care of uncertainty in missing values.

MICE assumes that the missing data are Missing at Random (MAR), which means that the probability that a value is missing depends only on the observed values and can be predicted using it. It imputes data on a variable by variable basis by specifying an imputation model per variable.

The mice package works analogously to PROC MI/ PROC MIANALYZE. The mice() function performs the imputation, while the pool() function summarizes the results across the completed data sets. The method option to mice() specifies an imputation method for each column in the input object.

In the analysis below previously generated random INDS dataset will be used.

```
#Load data
> library(haven)
> inds <- read_sas("D:/Downloads/inds_tr.sas7bdat", + NULL)
> input <- inds_tr
#Get summary
> summary(input)
# Load 'mice' package
> install.packages("mice")
> library(mice)
# Using mice for looking at missing data pattern
> md.pattern(input)
```

A more helpful visual representation can be obtained using, for example, VIM package.

The following problem was faced: since the change from baseline should be used as a covariate in a fit model, a new column should be added to the object of mids(Multiply Imputed Data Set) type after imputation. To do so a "passive" imputation should be applied.

```
> set.seed(123)
> chg6 <- NA
> inds_tr2<- cbind(inds_tr, chg6)
> inds_tr3 =
inds_tr2[c("subjid","trtn","vis1","vis2","vis3","vis4","vis5","vis6","chg6")]
> ini <- mice(inds_tr3, maxit = 0)
> meth <- ini$meth
#set "chg6" to passive imputation
```

```
> meth["chg6"] <- "~ I(vis6 - vis1)"
> imputed_Data <- mice(inds_tr3, m=5, maxit = 50, meth = meth, seed = 500)
```

Depending on how big the data set is and how many iterations are going to be performed, this can take a while (from 30 sec to even a few hours).
A short explanation of the parameters used:

1. m – Refers to 5 imputed data sets
2. maxit – Refers to the number of iterations taken to impute missing values
3. method – Refers to the method used in imputation. A predictive mean matching method was used in this case(it is a default method). Other imputation methods can also be used, typing methods(mice) will show a list of the available imputation methods.

```
#check imputed values(currently checking given variable vis1)
> imputed_Data$imp$vis1
#check what methods were used for imputing
> imputed_Data$method
```

subjid	trtn	vis1	vis2	vis3	vis4	vis5	vis6	chg6
“”	“”	“pmm”	“pmm”	“pmm”	“pmm”	“pmm”	“pmm”	“~ I(vis6 – vis1)”

Table 7 Imputation methods

SUBJID and TRTN have no method since there were no missing values for these variables, for CHG6(change from baseline on week 8) method was specified manually, for all the other variables method is equal to “pmm”.
pmm stands for Predictive Mean Matching and this is a basic/default imputation method used by MICE package.
Other imputation methods can be used as well, type > methods(mice) for the list of available imputation methods.
Predictive mean matching calculates the predicted value of the target variable Y according to the specified imputation model. For each missing entry, the method forms a small set of candidate donors (typically with 3, 5 or 10 members) from all the complete cases that have predicted values closest to the predicted value for the missing entry. One donor is randomly drawn from the candidates, and the observed value of the donor is taken to replace the missing value. It is assumed that the distribution of the missing cells is the same as the distribution of the observed data of the candidate donors.
Imputations are based on values observed elsewhere, so they are realistic. Imputations outside the observed data range will not occur, thus evading problems with meaningless imputations (e.g., negative body height).
Since there are 5 imputed data sets, any could be selected using the complete() function.

```
#get complete data (for example 2nd out of 5)
> completeData <- complete(imputed_Data,2)
> head(completeData)
```

The next step in the analysis is to fit a linear model to the data. A reasonable question would be what imputed dataset to choose. The mice package makes it again very easy to fit a model to each of the imputed datasets and then pool the results together
Creating a regression model taking VIS1 as a predictor variable and CHG6 as a response variable taking into account the interaction between TRTN(Treatment) and VIS1(Baseline).
After Multiple Imputation has been performed, the next steps would be to apply statistical tests in each imputed dataset and to pool the results to obtain summary estimates
The pool() function combines the estimates from m repeated complete data analyses. A typical sequence of steps to carry out a multiple imputation analysis is as follows:

1. Impute the missing data by the mice function, resulting in a multiple imputed data set (class mids);
2. Fit the model of interest (scientific model) on each imputed data set by the with() function, resulting an object of class mira;
3. Pool the estimates from each model into a single set of estimates and standard errors, resulting in an object of class mipo;
4. Optionally, compare the pooled estimates from different scientific models by the pool.compare() function.

A common error is to reverse steps 2 and 3, i.e., to pool the multiply-imputed data instead of the estimates. Doing so may severely bias the estimates of scientific interest and yield incorrect statistical intervals and p-values. The pool() function will detect this case.

```
# Create An Analysis Of Variance Model.
> fit <- with(data = imputed_Data, exp = lm(chg6 ~ vis1+ trtn))
```

"*" is used for models with interaction between the categorical variable and the predictor variable
"+" is used for models without interaction between the categorical variable and the predictor variable

TRTN	Treatment group	Categorical Variable
VIS1	Baseline	Continuous Predictor Variable
CHG6	Change from baseline	Response Variable

Table 8 Model elements description

```
#combine results of all 5 models
> summary(pool(fit), conf.int = TRUE)
```

Remember that the mice function was initialized with a specific seed, therefore the results are somewhat dependent on this initial choice. To reduce this effect, a higher number of datasets could be imputed, by changing the default m=5 parameter in the mice() function.

RESULTS COMPARISON

Eventually the results could be compared.
As stated above the MIXED procedure doesn't use missing values, so for the results without any imputation a complete case analysis is performed.

Number of Observation Read	500
Number of Observation Used	440
Number of Observation Not Used	60

Table 9 Observations usage

Table 10 shows statistical results before and after the multiple imputation.

Statistic	Result					
	SAS without MI			SAS with MI		
P-value	0.2852			0.3138		
LS Means	Estimate	Lower CL	Upper CL	Estimate	Lower CL	Upper CL
Treatment A	-0.08189	-0.4459	0.2821	-0.098242	-0.44103	0.244550
Treatment B	0.1942	-0.1585	0.5468	0.145027	-0.18625	0.476304
LS Diffs	Estimate	Lower CL	Upper CL	Estimate	Lower CL	Upper CL
	0.2760	-0.2310	0.7831	0.243269	-0.23017	0.716704

Table 10 SAS Results comparison

P-value - P-value for the overall treatment effect
LS Means - Combined LSM estimates of change from baseline in Sleep Quality for each treatment group
LS Diffs - Combined LSM estimates of difference between Treatment A and Treatment B in the change from baseline in Sleep Quality

Statistic	Result	
	R without MI	R with MI
P-value	0.285	0.349
Lower CL	-0.193	-0.259
Upper CL	0.314	0.729

Table 11 R Results comparison

In this particular case we can say that p-value is much higher than the level of significance(0.05) regardless of imputation, but as we can see p-value, lsmeans and lsdiffs vary significantly before and after performing the multiple imputation. Using another set of data might have boundary results and performing the multiple imputation will play critical role in accepting or rejecting of null hypothesis.

CONCLUSION

Missing data will always be a limitation when interpreting trial results; even if the data are MCAR, the missing data will result in loss of statistical power. These limitations due to missing data should always be thoroughly considered and discussed by the trialists. As always, prevention is better than cure. To mount professional prevention, trials need to be focused and pragmatic. Trial results based on data with missing values should always be interpreted with caution. It is not possible to differentiate between MAR and MNAR so the validity of the underlying assumptions behind, for example, multiple imputation may always be questioned, and when the data are MNAR, no methods exist to handle missing data appropriately.
Missing data reduces the power of a trial. Some amount of missing data is expected, and the target sample size is increased to allow for it. However, it cannot eliminate the potential bias. More attention should be paid to the missing data in the design and performance of the studies and in the analysis of the resulting data.
The best solution to the missing data is to maximize the data collection when the study protocol is designed and the data are collected. Application of the sophisticated statistical analysis techniques should only be performed after the maximal efforts have been employed to reduce missing data in the design and prevention techniques.
In this paper the most commonly used ways were explored to check for missing values as well as to complete appropriate missing values in both SAS and R. They provide convenient methods for multiple imputation and further analysis of completed data sets. The analyst can select from a variety of imputation models and successfully apply them to different kinds of missing data patterns. Both SAS and R have flexible and user friendly tools which can go through all the stages of complex multiple imputation.

REFERENCES

Rubin, D.B. 1987. Multiple Imputation for Nonresponse in Surveys. New York: John Wiley & Sons, Inc.

Rubin, D.B. 1996. “Multiple Imputation After 18+ Years.” Journal of the American Statistical Association 91: 473-489.

Allison, P.D. (2000), “Multiple Imputation for Missing Data: A Cautionary Tale,” Sociological Methods and Research, 28, 301–309

Horton, N.J. and Lipsitz, S.R. (2001), “Multiple Imputation in Practice: Comparison of Software Packages for Regression Models With Missing Variables,” Journal

Kenward MG. The handling of missing data in clinical trials. Clin. Investig. (Lond.) 3, 241–250 (2013).

Kenward MG, Carpenter JR. Multiple imputation. In: Longitudinal Data Analysis: A Handbook of Modern Statistical Methods. Davidian M, Fitzmaurice G, Verbeke G, Molenberghs G (Eds). Chapman & Hall/CRC, London, UK, 477–500 (2008).

Berglund, Patricia and Heeringa, Steven, 2014. Multiple Imputation of Missing Data Using SAS® . Cary, NC: SAS Institute Inc

Little, R. J. A., & Rubin, D. B. (2002). Statistical analysis with missing data. Hoboken, NJ: John Wiley and Sons, Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Please contact the author at:

Author Name: Vadim Kalinichenko
Role: Senior Statistical Programming Analyst
Company: Intego Group, LLC
Address: 43/2 Gagarin Avenue
City / Postcode: Kharkov 61001, Ukraine
Work Phone: +38 057 755 7020 ext. 2425
Fax: +38 057 728 30 35
Email: vadim.kalinichenko@intego-group.com
Web: https://intego-group.com