

Sensitivity Analysis for Missing Data Using Control-based Pattern Imputation

Jun Feng, Seagen Inc.
Jingmin Liu, Seagen Inc.

ABSTRACT

Randomized controlled clinical trials are known to be an effective way to minimize bias and draw convincing conclusions on the efficacy and safety of a drug, but the data quality and the statistical methods used for the analyses will highly influence the results. Inevitable scenarios such as protocol deviations and subjects lost to follow-up will lead to missing data. How the missing data are handled is crucial for the integrity of the statistical analysis, especially for efficacy endpoints. Sensitivity analysis is a useful method to stress-test the credibility of statistical conclusions and explore the impact of the missing records. Using imputation methods to fill in these missing observations and consider imputation simulation results through sensitivity analysis will render a more robust statistical conclusion. Comparing these outcomes between the original and the imputed data can highlight the influence of the missing data on the results and establish credibility of the conclusions.

SAS® provides an efficient way for such sensitivity analyses using the MI and MIANALYZE procedures. This paper illustrates the statistical background and implementation difference between MCAR (Missing Completely at Random) and MNAR (Missing not at Random) assumptions along with an example of data manipulation for monotone missing data before sensitivity analysis, using Control-based Pattern Imputation with Mixed Model in SAS, and how to compare and interpret the sensitivity statistical analysis results.

INTRODUCTION

Missing data is inevitable even with the most careful planning and rigorous implementation of randomized clinical trials. The degree, pattern, and underlying cause of missingness will have great impact on the results and their interpretation. Dealing with missing data is an active research area with a wealth of statistical methods developed over the years. However, the application and implementation of well-established statistical methods to address missing data problem is scarce.

There are many types of missing data patterns; various assumptions could be made towards these missing data. This paper focuses on monotone missing data, briefly discusses two different missing data assumptions, touches upon statistical methods that consider these assumptions, and illustrates the imputation of missing data. In the end, results of using the original unimputed data will be compared with the analysis results using the imputed data.

METHOD

One type of commonly seen missing data patterns in clinical trials is monotone missing. This occurs when an observation is first missing for an individual at certain visit, then all the observations for this individual are missing at subsequent visits. In the clinical trials setting, this pattern is often associated with subject drop out, study discontinuation due to futility, or an adverse event, and therefore has great implication on the study results.

This paper considers two common missing data assumptions: missing completely at random (MCAR) and missing not at random (MNAR). For MCAR, it is assumed that the subjects with missing data are a random sample of all the subjects enrolled in the trial. The missing data are unrelated to any study variables or activities. For MNAR, it is assumed that the missing data is related to some values of study variables. In the context of monotone missing data, one example of monotone MCAR is subjects dropping out randomly across treatment arms, unrelated to any treatment assignment; one example of monotone MNAR is study discontinuation due to adverse effects of a certain study treatment where a higher discontinuation rate is observed in one of the study treatment arms.

MCAR is a strong assumption. Randomization and intent-to-treat analyses provide some degree of protection against violation of this assumption, hence a primary analysis is often conducted assuming missing at random. However, sensitivity analysis under the MNAR assumption should be performed to examine the robustness of results. There are various statistical methodologies for imputing monotone missing and drop-out. This paper follows the approach established in Little & Yao 1996. Instead of assuming the monotone missing (drop-out) on an “as treated” model, this paper assumes these dropouts on an “as control” model. The latter is believed to be the more likely scenario, especially for large confirmatory trials when treatment is still under investigation, and it will be highly unlikely for patients who drop out of the clinical trials to obtain the experimental drugs. It is reasonable to assume that the monotone missing pattern will be more in line with what is seen in the control, when patients resume the standard-of-care after dropping out from the clinical trials.

This ‘control-based’ imputation method will be implemented in the framework of pattern-mixture models where the pattern of control arms will be used to identify or fill in the monotone missing data. This process will be demonstrated through an example data set.

EXAMPLE DATA

Demonstration of the imputation method will use an example data set with continuous efficacy variables. Details are as below:

- SUBJID – Subject ID
- TRTN – Numeric treatment variable, 1 as treatment and 0 as placebo
- VISITNUM – Numeric visit variable, possible values are 1, 29, 57, and 85
- AVAL- Continuous visit-level efficacy endpoint variable (e.g., sum of the products of diameters (SPD) in an oncology study)
- BASE – Equals AVAL value at the baseline visit, this is populated at the subject level
- CHG – Equals AVAL – BASE for each visit after baseline

Each subject will have 4 visits even if values are missing for certain visits. The total number of subjects in the example data set is 107. Figure 1 shows data for one subject.







 SUBJID	 TRTN	 VISITNUM	 AVAL	 BASE	 CHG
001	1	1	3.66	3.66	
001	1	29	0.768	3.66	-2.892
001	1	57	0.73	3.66	-2.93
001	1	85	0.65	3.66	-3.01

Figure 1: Sample data

Table 1: Number and Percentage of Missing AVAL by Treatment and Visit

Visit	Treatment Arm	Placebo Arm
Baseline	0	0
29	4 (6.9%)	10 (20.41%)
57	10 (17.24%)	13 (26.53%)
85	14 (24.14%)	19 (38.78%)

Table 1 shows the count and percentage of missing AVAL. As we can see, later visits usually have a larger percentage of missing values, and in this example data, the placebo arm has more missing values than the treatment arm.

CONTROL-BASED PATTERN IMPUTATION PROCESS

In this section we will demonstrate how to perform imputation and sensitivity analysis on our example data. The idea of control-based pattern imputation is that for those subjects lost to follow up, we consider them as still in trial and taking the placebo treatment. Using the placebo arm as reference to impute the missing value will be more reasonable than other traditional methods of imputation, which still assume the subject is under the treatment effect. More details for programming are in the example below.

As we mentioned in the Method section on the definition of MCAR and MNAR, under the assumption of MCAR, we use SAS programming procedure PROC MI and PROC MIANALYZE for the imputation process. Firstly, we need to identify the analysis procedure to use and get the analysis results on original data. In this example, PROC MIXED is applied and our analysis goal is comparison of mean change from baseline between two treatment arms. This analysis result can be used later in comparison with results of the imputed data under the assumption of MNAR.

Although the focus of this paper is on monotone missing, it is unavoidable that the data will have some non-monotone missing as well. We will first identify and impute the intermittent missing data to make the data monotone, then we can handle the monotone missing as described in previous section. In PROC MI there are effective ways to handle both types of missing data. The MCMC method can be used for non-monotone missing records, and the monotone option in PROC MI can help us handle monotone missing records.

In this demonstration we use both the MCMC method and monotone options in PROC MI. Detailed procedures are as follows:

1. Use the MCMC method in PROC MI to impute the intermittent missing records to make the data monotone missing. A simulation of imputed values is also generated, which can be used later in PROC MIANALYZE to combine and get the final estimate.
2. Use the monotone option to impute remaining monotone missing values one visit to the next, repeat the procedure until all the records are imputed. To utilize control-based pattern imputation we need to include all the placebo arm records in each imputation; for the treatment arm only those that need to be imputed in the current visit are included.
3. Run PROC MIXED on imputed data and output the statistical result; use PROC MIANALYZE to combine all imputation simulations and generate the mean change from baseline for imputed data.
4. Use PROC MIXED on original data and compare with the results from imputed data.

Before we use the MCMC method to impute the intermittent missing values, it is convenient to set up a pointer variable, say LASTWK (last non-missing visit). This can be useful when we identify monotone

missing subjects at the current visit level when we are doing step 2 mentioned above and impute them. In this demonstration we use a simple macro to transpose the data as needed and set up LASTWK.

```
%macro select(visitnum=);
  data eff&visitnum(keep=subjid trtn d28_&visitnum base);
  set eff_f;
  if visitnum=&visitnum then do;
    d28_&visitnum=aval;
    output;
  end;
  run;
%mend;

data eff_h;
  merge eff1 eff29 eff57 eff85;
  by subjid;

  if d28_85 ne . then lastwk=85;
  else if d28_57 ne . then lastwk=57;
  else if d28_29 ne . then lastwk=29;
  else lastwk=1;
  run;
```

After that we can start imputing the intermittent missing subjects. Option NIMPUTE defines the total simulation number. In the VAR statement we include all variables that need to be imputed.

```
proc mi data=eff_h out=eff_h_mono nimpute=100 seed=12345;
  var trtn base d28_29 d28_57 d28_85;
  mcmc chain=multiple impute=monotone;
  run;

proc sort data=eff_h_mono nodup;
  by _imputation_;
  run;
```

Once we finish the intermittent imputation, we can get the monotone missing data. Next, we can impute them visit by visit, use variable LASTWK to control as a pointer variable. As we mentioned before regarding the logic of our control-based imputation method, we need to make sure that all the control-arm subjects and all subjects that need to be imputed at the current visit are included in the imputation simulation. As described in step 2, ensuring that we include all control arm subjects in each imputation cycle is critical for this control-based imputation process.

```
%macro split_impute(inputs=, visitnum=, varlist=);

  data eff_h_mono_imp&visitnum eff_h_mono_rest&visitnum;
  set &inputs;
  if trtn=1 and lastwk>=&visitnum. then output eff_h_mono_rest&visitnum;
  else output eff_h_mono_imp&visitnum; /*Make sure that all the control-
  arm subjects, all the subjects need to be imputed at current visit level are
  included */
proc mi data=eff_h_mono_imp&visitnum out=eff_h_reg_imp&visitnum nimpute=1
  seed=12345;
  by _imputation_;
```

```

var &varlist;
monotone reg(d28_&visitnum);
run;

data eff_h_imp&visitnum;
set eff_h_reg_imp&visitnum /*Subjects that are imputed*/
eff_h_mono_rest&visitnum /*Subjects that are not imputed and just carry
to the next step*/;
run;

proc sort data=eff_h_imp&visitnum;
by _imputation_;
run;

%mend;

/*Repeat imputation visit by visit and carry the final data sets to the next
visit*/
%split_impute(inputds=eff_h_mono, visitnum=29, varlist=%str(base d28_29));
%split_impute(inputds=eff_h_imp29, visitnum=57, varlist=%str(base d28_29
d28_57));

```

After the visit-by-visit imputation and transpose, we have the imputed data and can start on the model programming. PROC MIXED is used in this example. PROC MIXED can handle the simulation data with statement “by _imputation_”. The program for this modeling is as shown below.

```

data effx_vertical;
set eff_vertical;
trt=1-trtn;
run;

proc mixed data=effx_vertical method=reml;
by _imputation_;
class visitnum trt subjid;
model chg_spd=base trt visitnum visitnum*trt /ddfm=kenwardroger;
repeated visitnum/subject=subjid type=un;
lsmeans trt*visitnum / at means diff cl;
ods output lsmeans=lsmeans_sim diffs=diffs_sim;
run;

```

Output dataset LSMEANS_SIM (example in figure 2) stores the model effect by each level. Data set DIFFS_SIM (example in figure 3) stores the effect comparison between two treatment arms. We need to filter the meaningful comparisons which target the same visit level.









	 _IMPUTATION_	 EFFECT	 VISITNUM	 TRT	 ESTIMATE	 LOWER	 UPPER	 PROBT
1	1	VISITNUM*TRT	29	0	-20.9521	-22.4994	-19.4048	<.0001
2	2	VISITNUM*TRT	29	0	-21.1160	-22.6735	-19.5585	<.0001
3	3	VISITNUM*TRT	29	0	-21.1708	-22.7099	-19.6316	<.0001
4	4	VISITNUM*TRT	29	0	-21.1280	-22.6810	-19.5750	<.0001
5	5	VISITNUM*TRT	29	0	-21.1892	-22.7377	-19.6407	<.0001
6	6	VISITNUM*TRT	29	0	-20.9659	-22.5218	-19.4100	<.0001
7	7	VISITNUM*TRT	29	0	-20.9952	-22.5495	-19.4408	<.0001

Figure 2: Model Effect by Imputation











 _IMPUTATION_	 EFFECT	 VISITNUM	 TRT	 _VISITNUM	 _TRT	 ESTIMATE	 LOWER	 UPPER	 PROBT
1	VISITNUM*TRT	29	0	29	1	1.0533	-1.2333	3.3398	0.3629
2	VISITNUM*TRT	57	0	57	1	1.3929	-0.7685	3.5543	0.2039
3	VISITNUM*TRT	85	0	85	1	0.1734	-0.6882	1.0350	0.6907
4	VISITNUM*TRT	29	0	29	1	0.8650	-1.4366	3.1666	0.4575
5	VISITNUM*TRT	57	0	57	1	1.2411	-0.9880	3.4702	0.2719
6	VISITNUM*TRT	85	0	85	1	0.3469	-0.6297	1.3236	0.4827

Figure 3: Meaningful Comparison of Treatment Effect at Same Visit

Once we get the model effect and difference data, we can use PROC MIANALYZE to combine all the simulation results and have an integrated model result. LSMEANS_SIM and DIFFS_SIM generated from the previous step are used here to combine the combined estimates and confidence intervals.

```
proc mianalyze data=lsmeans_sim;
  by visitnum trtn;
  modeleffects estimate;
  stderr stderr;
  ods output parameterestimates=parameterestimates;
run;
```

```
proc mianalyze data=diffs_sim;
  by visitnum;
  modeleffects estimate;
  stderr stderr;
  ods output parameterestimates=parameterestimates;
run;
```

Figure 4 shows the model result from the original data without any imputation; figure 5 shows the imputed model result. MEANA is the mean of treatment arm, CIA is the confidence interval of the treatment arm, MEANP is the mean of placebo, MEAND is the mean difference of treatment - control, and PVAL stands for the P value of the significance of the mean difference between the two treatment arms.









 VISITNUM	 MEANA	 CIA	 MEANP	 CIP	 MEAND	 CID	 PVAL
1	29 -21.05 (0.8249)	(-22.69, -19.41)	-21.95 (0.9716)	(-23.88, -20.02)	0.90 (1.2745)	(-1.63, 3.43)	0.4811
2	57 -21.87 (0.7888)	(-23.44, -20.30)	-23.22 (0.9281)	(-25.07, -21.38)	1.35 (1.2180)	(-1.07, 3.77)	0.2698
3	85 -23.34 (0.2894)	(-23.92, -22.76)	-22.98 (0.3509)	(-23.68, -22.28)	-0.36 (0.4541)	(-1.27, 0.54)	0.4258

Figure 4: Statistical Analysis Output for Original Data









 VISITNUM	 MEANA	 CIA	 MEANP	 CIP	 MEAND	 CID	 PVAL
1	29 -21.15 (0.8556)	(-22.83, -19.47)	-22.28 (0.9853)	(-24.21, -20.35)	1.21 (1.2419)	(-1.22, 3.65)	0.3298
2	57 -21.95 (0.8173)	(-23.55, -20.35)	-23.39 (0.9412)	(-25.24, -21.55)	0.72 (1.2390)	(-1.71, 3.15)	0.5634
3	85 -23.33 (0.2614)	(-23.84, -22.81)	-23.08 (0.3074)	(-23.68, -22.48)	-0.24 (0.4003)	(-1.03, 0.54)	0.5420

Figure 5: Statistical Analysis Output for Imputed Data

As expected, the imputed results differ from the results based on the original, unimputed data. The difference is smaller for earlier visits when the percentage of dropouts is small. Towards the end of the study, as the missing pattern is imputed using controls, the treatment mean was brought towards the control, shrinking down the difference between treatment and placebo. Imputation results, using the entire dataset, were able to offer a tighter confidence interval, increasing the precision. However, despite these differences, the overall conclusion and statistical significance stay the same (i.e., non-significant). The sensitivity analysis under a reasonable assumption for missing data stress-tested the primary analysis, with both arriving at the same conclusion.

CONCLUSION

The paper uses an example data set to demonstrate one way of conducting a sensitivity analysis for monotone missing data under an MNAR assumption. Control-based pattern imputation is an effective sensitivity analysis method to check the validity of the statistical conclusion. But we also see that there could be some limitation of this method. As a method that relies on placebo arm records as an imputation reference, if there are too many missing values in the control arm it may influence the imputation results. We hope that through this simple demonstration people can have a better understanding of sensitivity analysis and also bring more attention to the statistical methods research on missing data and practical implementation.

REFERENCES

- [1] SAS Institute Inc. 2014. SAS/STAT®13.2 User's Guide. Cary, NC: SAS Institute Inc.
- [2] Ratitch, B, O'Kelly, M, Implementation of Pattern-Mixture Models Using Standard SAS/STAT Procedures. PharmaSUG Conference, 2011.
- [3] Smith, C, Kosten S, Multiple Imputation: A Statistical Programming Story. PharmaSUG Conference, 2017.
- [4] Little, R., Yau, L. Intent-to-Treat Analysis for Longitudinal Studies with Drop-Outs. Biometrics, 1996, vol. 52, 1324-1333.

ACKNOWLEDGMENTS

We want to thank Seagen Inc. for the support on this work. We also want to thank Bala Pitchuka and Johnny Maruthavanan for their reviews, comments and support.

CONTACT INFORMATION

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

Jun Feng
Seagen Inc.
Email: jfeng@seagen.com

Jingmin Liu
Seagen Inc.
Email: jiliu@seagen.com

SAS® and all other SAS® Institute Inc. product or service names are registered trademarks or trademarks of SAS® Institute Inc. in the USA and other countries. ® indicates USA registration.