Comparing SAS® PROC MI and IVEware Callable Software

Bruno Vizcarra and Amang Sukasih
Mathematica Policy Research
1100 1st Street NE, Washington, DC

ABSTRACT

Multiple imputation has become a common technique for dealing with missing data, to account for the variability involved with imputing data values. Advancements in technology have allowed the development of a number of software and modules/packages that can perform multiple imputation in SAS, R, Stata, and other programs. In this paper we focus on two software packages: SAS PROC MI procedure (SAS Institute 2000) and IVEware. IVEware (Raghunathan et al. 2002) is an imputation and variance estimation software that can be run in SAS (SAS callable). SAS PROC MI provides imputation options including regression imputation, Markov chain Monte Carlo (MCMC) techniques, and fully conditional specification, whereas the IVEware implements a sequential regression imputation technique. Although the basic modeling and prediction in SAS PROC MI and IVEware are comparable; the imputations are developed under different distribution assumptions of variables with missing values. This paper compares the two methods using a data set with count variables having a Poisson distribution. We compare the imputed data with two approaches. The first approach directly imputes the count variable; the second is a modified approach where a binary variable indicating zero or non-zero count is imputed and then the non-zero counts are imputed. We discuss the limitations and issues encountered.

Keywords: missing values, item nonresponse, missing at random, multiple imputation, sequential regression imputation

INTRODUCTION

Missing data is a common problem in survey data that can influence data analysis. Reasons for missing data can range from the poor design of a study (for example, confusing wording in questions), data entry errors, or participation denial, to truly random missing values, such as a participant skipping a question unintentionally. Missing data can be classified into three types (Rubin 1987): missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR). MCAR indicates that the probability of a missing value in the data set does not depend on the value of any other variable, either known or unknown, in the data set. MAR indicates that the probability of such a value being missing may be related to the values of other variables. MNAR indicates the probability that a missing value is associated with the missing variable itself and with other variables in the data set.

Methods to deal with missing data can include **complete case analysis**, where only observations that have no missing data in any variable are used, or **pairwise deletion**, where specific variables with no missing data are used in some of the analyses but not in others; however, these methods often have serious disadvantages due to information loss from dropping cases with missing values. These approaches also ignore possible differences between the complete cases and the incomplete cases that could bias the data; the resulting inference might not be applicable to the population of all cases, especially if there are only a small number of complete cases. Using only cases with no missing values also makes a strong assumption that the data is MCAR—which may not always be the case when data has missing values. Other methods, like **single imputation**, replace the missing value with either a mean value or another appropriate value from a similar unit or "neighbor," as in hot deck imputation. These approaches, though more statistically acceptable, treat the imputed values as true values, not taking into account the error that imputation introduces into the analysis; the variance of estimates computed based on imputed data is believed to usually be underestimated (Rubin 1987).

Multiple imputation, first proposed in the 1970s and developed in the 1980s, improves upon the traditional methods and has become a popular method for dealing with missing data. It maintains the advantages of single imputation—such as the ability to use complete case techniques on a complete data set, using observed data to impute missing values, and maintaining the consistency of answers in the data set—while also adding the uncertainty of the imputed values to the calculation of variance of estimate. Multiple imputation replaces missing values with a list of possible values, generating a user-defined number of multiple data sets. The data sets are then analyzed independently and the results combined.

Advances in technology have allowed the development of a number of software and modules/packages that can perform multiple imputation. Some of these include PROC MI/MIANALYZE in SAS; SAS callable IVEware; MICE,

argImpute, and Amelia II in R; ICE in STATA; Schafer's NORM, MIX, CAT, and PAN packages in S-PLUS; and stand-alone software such as SOLAS by Statistical Solutions and LogXact by Cytel studio.[1] This paper focuses on two approaches available using SAS. The first is SAS's own developed procedure, PROC MI (SAS Institute 2000). The second is IVEware (Raghunathan et al. 2002), a SAS callable macro that performs multiple imputation in the SAS system but in a sequential manner, allowing the user to implement other features to control the imputation model.

In comparing SAS PROC MI with IVEware, our variable of interest is a count (discrete variable), which distributes as a Poisson random variable. However, SAS PROC MI does not have an option to model such distribution; rather, it approximates it by assuming a normal distribution. IVEware allows more options of type of imputed variables and can model a Poisson distribution for a variable such as count. For this comparison, a synthetic data set was generated with complete case categorical variables and a count response variable with zero and missing values. We used these two packages to deal with missing values in a count variable where zero count value is allowable in the variable and the missing mechanism is believed to be MAR.

MULTIPLE IMPUTATION

Multiple imputation has been an accepted approach to dealing with missing data since the 1970s, when the idea was first introduced by Rubin (1977). This approach takes into account the uncertainty of imputed values when analyzing the imputed data, while retaining the advantages of single imputation. The concept of multiple imputation uses fitted estimates for the mean and correlation matrix and the standard error as parameters to build a Bayesian posterior distribution from which values are drawn until the imputed values are stabilized.

Given the ability of computers to process large, complex data sets, multiple imputation has become an accessible approach to dealing with missing data.

Multiple imputation usually involves a three-step process:

1. **Imputation**. Generate a set of plausible values for the missing observations. These plausible values are sampled from their predictive distribution based on observed values.
2. **Analysis**. Perform the desired analysis on each set of generated data using complete case techniques. Results on each data set will vary due to the difference in values during the multiple imputations.
3. **Combination**. Combine the results from all the analyzed data sets. Combination will take the average of the results in step 2. Standard errors are calculated using Rubin's rules (Rubin 1987), which aggregate the average variance within imputed data sets and the variance between the multiple estimates from multiple imputed data.

The imputation stage relies on assumptions regarding the missing data, most specifically, the MAR assumption, where the distribution of imputed variable is modeled with other (nonmissing) variables used as the covariates/predictors. Between three to five imputations are adequate in multiple imputation (Rubin 1996).

The theory behind multiple imputation is to use the observed data available to build a Bayesian posterior distribution where means, covariance, and standard errors are used as parameters for the imputation model. Following Bayes' Theorem, a parametric model is compounded for the observed data, the unknown model parameters are modeled with prior distributions, and a defined number of independent draws are simulated from the conditional distribution of the missing data given the observed data. Each draw also generates a random error component, so results are not the same across imputed data sets (Rubin 1987).

[1] For more detailed information on other available software, see http://www.math.smith.edu/~nhorton/muchado.pdf.

SAS PROC MI

SAS PROC MI performs the imputation stage and can be used with either monotone or non-monotone missing patterns[2]. For monotone patterns, the MONOTONE statement is available under PROC MI. This statement can implement different imputation modeling methods according to the type of variable being imputed. For continuous variables, the REG method, using a regression model, and the REGPMM method, using a predictive mean matching method, are available. For categorical variables, the LOGISTIC method, which uses logistic regression modeling, and the DISCRIM method, which only allows continuous variables in the imputation model, are available. The PROPENSITY method can be used for both continuous and categorical variables and uses a propensity score to impute data. For the non-monotone pattern, a Markov chain Monte Carlo (MCMC) statement (Schafer 1997) that assumes multivariate normality or a fully conditional specification (FCS) statement (van Buuren 2006) that assumes the existence of a joint distribution for all variables are available. The MCMC statement can also be used in a hybrid model where the data set is separated into monotone and non-monotone subsets; the MCMC imputes the non-monotone data until it has a monotone pattern, then the regression method is used for the monotone component. The FCS statement also has specific modeling approaches given the type of variable imputed. The LOGISTIC, DISCRIM, REG, and REGPPM methods in the MONOTONE statement are also available in the FCS statement.[3]

The FCS statement is a new addition to the PROC MI in SAS version 9.3 and its currently an experimental statement. The method does not start with a specified multivariate posterior distribution of observed data, but instead uses a separate conditional distribution for each imputed variable. A two-step process is performed for each imputation: (1) fill-in and (2) imputation. In the fill-in step, the missing values for all variables are filled in sequentially over the variables taken one at a time, providing starting values for these missing values for the second step (imputation). In the imputation step, the missing values are imputed sequentially over the variables taken one at a time at each iteration for the number of iterations specified (van Buuren 2006).

Other features in PROC MI include the available transformation and back-transformation of variables, given that the REG and MCMC methods assume a multivariate normal distribution of the imputed variable (Schafer 1997), as well the ability to specify a minimum and maximum value for imputed values for one or all variables and a rounding option for imputed values. The FCS method assumes the existence of a joint distribution for all variables.

The general coding procedure for PROC MI using FCS method is as follows:

```
PROC MI DATA=DATAIN MINIMUM=MINIMUMVALUES NIMPUTE=N OUT=DATAOUT;
       TITLE "PROC MI, GENERIC CODE FOR FCS REGRESSION";
       FCS REG(Y = X1-X3 / DETAILS);
       VAR Y X1-X3;
RUN;
```

MINIMUM is the command for a minimum value to be imputed for the variables, where you can specify a number for all the variables being imputed or only for certain ones. NIMPUTE will indicate to SAS how many replicates to produce. The FCS approach requires a specification of what method to be used—in this case, the regression method (REG) with a response variable *Y* to be imputed and covariates X_1–X_3 to be used for the regression model. The DETAILS option displays the regression coefficients in the regression model used in each imputation. If only certain variables are to be imputed and/or used in the imputation model, a VAR statement is needed, specifying the desired variables.

Once the PROC MI procedure has been completed, complete case analysis procedures can be performed on each full data set. The MI procedure generates a variable labeled *_imputation_* during the MI step, with *n* values corresponding to however many replicate data sets were requested. For our example, we calculate the mean of our example variable *Y* using PROC MEANS. The standard error for each calculation must also be obtained in order to accurately combine the results from the imputed data sets to obtain the final results. The general SAS coding for PROC MEANS for our example is as follows:

---

[2] Suppose a dataset has variables X_1 through X_p. A dataset has a monotone missing pattern if a variable X_i is missing for a particular individual, then all subsequent variables X_j, $j > i$, are missing for that individual. Alternatively, if variable X_j is observed, then all previous variables X_i, where $i < j$, must be observed.

[3] For more information about the methods and modeling specifics, refer to the SAS PROC MI website. http://support.sas.com/documentation/cdl/en/statug/63962/HTML/default/viewer.htm#mi_toc.htm.

```
PROC MEANS DATA=DATAOUT;
       VAR Y;
       BY _IMPUTATION_;
       OUTPUT OUT=DATAOUT_MEAN MEAN=Y_MEAN STDERR=Y_STDERR;
RUN;
```

The output data set *DATAOUT_MEAN* now contains the mean and standard errors for each of the imputed data sets for variable *Y*. To obtain the final mean combining the results from *n* replicates, we must use the PROC MIANALYZE[4] in SAS. Specification of the variable of interest, in this case *Y_MEAN*, is needed in the MODELEFFECTS statement, as well as an indication of the stored standard errors for such variable, *Y_STDERR,* in the STDERR statement.

```
PROC MIANALYZE DATA=DATAOUT_MEAN;
       MODELEFFECTS Y_MEAN;
       STDERR Y_STDERR;
RUN;
```

PROC MIANALYZE will take the average of the *n* calculated means to generate a final mean estimate. It will also calculate the within variance of each imputed value, adding it to the between variance for each imputed data set to calculate the final variance. Other output includes a 95 percent confidence interval for the estimate as well as a display of the minimum and maximum values in the data set replicates.

IVEware

IVEware is an SAS callable routine built using the SAS macro language along with a set of independent C routines. It performs multiple imputation using the sequential regression imputation method. IVEware, being SAS callable software, has the advantage that it can be used in conjunction with other procedures and data steps before and after the imputation process. The IMPUTE module can perform multivariate imputations for relatively complex data structures when the data are MAR. This module can impute different type variables, such as continuous, counts, categorical with two or more categories (dichotomous or polytomous), and semi-continuous variables.

IVEware implements the sequential regression multivariate imputation (SRMI) method as described in Raghunathan et al. (2001). The basic approach of IVEware is to create multiple regression imputations sequentially. Given observed values (covariates), the joint conditional density of multiple variables with missing values can be factored into an individual conditional density function for each variable; this individual density is then modeled through a regression appropriate for the variable type, such as continuous, binary, polytomous, count, or mixed. Imputation for missing value is then drawn from a posterior predictive distribution through these regression models.

IVEware also has useful features, such as a restriction to impute only to certain subpopulations, ability to include skip patterns, upper and lower bound values for imputation of variables, and a transfer feature to include variables not used in the imputation in the output data set. It also allows a minimum R-squared feature and a maximum number of predictors feature to be used when a stepwise selection is performed in the candidate group of covariates.

In order to use IVEware, modules must be installed in the computer used and a small modification to the SAS core program is also required.[5] To perform the IMPUTE procedure in IVEware, one must create and save a .set program and then call it through a %IMPUTE macro command. A generic .set program is as follow:

[4] For complete PROC MEANS and PROC MIANALYZE information, see the SAS 9.3 user's guide website at http://support.sas.com/documentation/.
[5] There is also a stand-alone software version of IVEware that performs certain procedures. More information is available at http://www.isr.umich.edu/src/smp/ive.

```
DATAIN Datain;
DATAOUT Dataout ALL;

CATEGORICAL X1 X2 X3;
COUNT Y;

TRANSFER variables not to be used in the imputation process (e.g., ID
variables);

BOUNDS Y (>=0);

RESTRICT Y (variable = value(s))

MULTIPLES n;

PRINT COEF;

RUN;
```

The DATAIN and DATAOUT commands indicate the data set to be used and then outputted with the imputed values. The ALL keyword will store all the replicates in the same output data set. The CATEGORICAL and COUNT commands specify the type of variables in the data set in our example. (One could specify other types of variables, such as MIXED and CONTINUOUS.) The TRANSFER command allows us to move variables from the *datain* to the *dataout* without them being used in the imputation process. The BOUNDS command will set a minimum imputation value; for this example, we set a minimum value of 0 for variable *Y*. The RESTRICT command will only impute values for a certain subpopulation. MULTIPLES will create *n* number of replicates. The PRINT command will output key statistics and model details from the imputation; in this example code, the regression coefficients (COEF).[6] Once the .set program is created, the IVEware macro in the SAS editor can be called with the statement:

```
%IMPUTE(NAME=Example.set, DIR="Location of Example.set program");
```

The macro will execute and the list and log windows in the SAS interface will be used for output results. This is also executable in batch mode. Once the imputation step has been completed, regular SAS procedures can be performed on the data set replicates and PROC MIANALYZE can be used to combine such results.

EMPIRICAL COMPARISON

We compare results from the PROC MI procedure in SAS to the IMPUTE model in IVEWARE using a simulated dataset. The simulated data contains a set of nine categorical variables used as covariates, labeled X_1 to X_9, ranging from three to seven categories. The response variable, labeled *count_it*, is of count nature and containing a large number of zero values. In our example, the data consists of 7,871 observations. The covariates do not have any missing values, whereas the count variable has 15.2 percent (1,194 cases) missing values. The count variable (nonmissing values) has 7.85 percent (618 cases) of zero value, 75 percent of counts have counts values less than 300, and there are also extreme count values (maximum value of 8,018), reflecting a highly skewed distribution. Figure 1 shows the distribution of the *count_it* variable.

[6] For more detailed information and a complete list of options, refer to the IVE Manual ftp://ftp.isr.umich.edu/pub/src/smp/ive/ive_user.pdf.

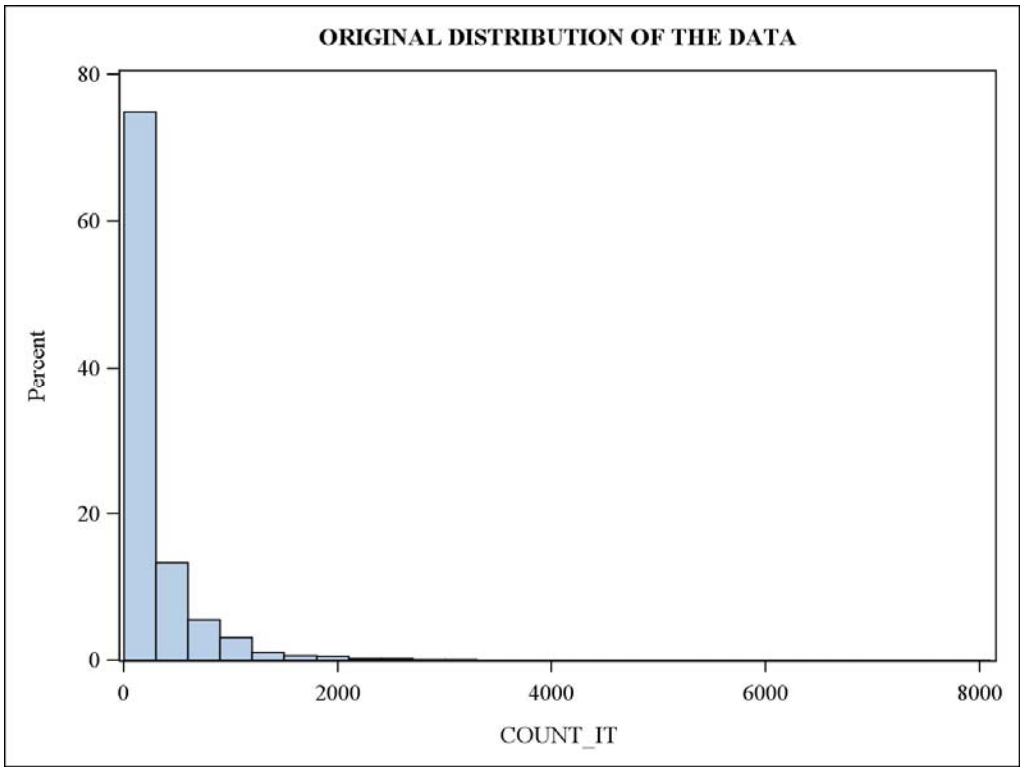

Figure 1. Empirical Distribution of Count Variable

The purpose of this example is to compare the counts imputed by PROC MI and IVEware and to estimate the mean value of the *count_it* variable. To assess the difference in imputation method, factors and features were kept as similar as possible in this example. We used the PROC MI's FCS method for comparison to the IVEware results since FCS uses a sequential approach to multiple imputation, similar to IVEware. Both PROC MI and IVEware include the option of imposing a minimum to the possible imputed values. For this example, the minimum value to be imputed is a zero count.

PROC MI's FCS procedure with the REG method assumes a multivariate normal distribution and will fit a normal regression model. In our example, the variable to be imputed is of count nature, following a Poisson distribution pattern. IVEware has the capability of fitting a Poisson regression model, but because PROC MI is not capable of doing so, the data had to be transformed as close as possible to a multivariate normal distribution. Research suggests a power transformation for a Poisson distribution variable (Anscombe 1948). In our example, a constant value was added to the count variable and the eighth square root was taken of this sum. This transformation was not needed for the IVEware approach since software specification of variable type—in this case, count variable—can be specified and a Poisson regression model can be created in IVEware.

The distribution of simulated count data is extremely skewed, as seen in Figure 1. We first ran imputation on this data and set a minimum imputed value of zero counts as a restriction.

The first imputation does not include the information needed to address the heavy zero distribution of the response variable. The number of zero counts imputed by PROC MI was very small, whereas the IVEware approach imputed none. Extra information could be useful to account for zero and non-zero counts to be imputed. In this case, an indicator imputation flag was created to implement a two-step process.

Since a heavy zero distribution is hard to address during imputation, this example compares an alternative approach to regular free-range imputation. The alternative approach was to conduct a two-step process, introducing a categorical imputation flag to address the heavy zero distribution. That is, the first step was to transform the missing variable into a binary variable indicating whether the count value is zero or greater than zero. This binary variable with

missing value was then imputed using the FCS approach with a logistic regression model. In this step, we essentially imputed the missing value with either zero or non-zero count one time.

The second step was to impute cases with non-zero count separately. That is, based on only cases with a value of one in the binary variable, we imputed counts for these cases with a restriction of imputed value to be greater than zero.

RESULTS

To keep with standard multiple imputation methodology, five replicates were generated for both approaches for each software method when imputing the response variable. In the second approach, which includes the imputation of a categorical flag variable as a first step, the categorical flag was only imputed once. The mean of the *count_it* variable was calculated and the results for both PROC MI and IVEware were combined using PROC MIANALYZE.

FIRST APPROACH

The first approach was to impute the response variable with only a minimum value restriction of zero. No extra information was added to the model to address the heavy zero distribution. For comparison purposes, the response variables from PROC MI and IVEware were renamed *count_mi1* and *count_ive1*, respectively.

The imputed results from PROC MI and IVEware in the first approach differ between the two software programs. Overall, the imputed distributions for all five replicates on each approach were similar. PROC MI imputed lower values in the lower end of the count distribution but a much larger maximum value than IVEware, influencing the mean and standard deviation of the overall distribution. Also, PROC MI imputed between four to nine cases with zero counts whereas the IVEware method did not impute any cases with zero counts in any replicate. Table 2 shows the distributions of imputed values in the first replicate for both *count_mi1* and *count_ive1* in the first approach

Table 1. Distribution of Both PROC MI and IVEware on First Approach, Replicate 1r

	Count_mi1	Count_ive1
Minimum	0	24
25th percentile	33	170
Median	101	198
75th percentile	248	312
Maximum	4,614	1,165
Mean	207.1	251
Standard Deviation	338.7	161.4
Number of 0 counts imputed	4	0

The estimated mean of *count_mi1* in PROC MI was lower than the IVEware estimate, but the combined variance was larger for PROC MI, probably due to the large maximum values imputed by PROC MI compared to IVEware. Table 3 shows the estimated minimum and maximum mean values encountered, as well as the between and within variances, along with the final variance of the combined results of the first approach for both PROC MI and IVEware.

Table 3. Combined Results for Both PROC MI and IVEware on First Approach

	Count_mi1	Count_ive1
Estimated mean	250.35	255.64
Between variance	0.75	0.004
Within variance	21.55	19.49
Combined variance	22.3	19.49

Looking at the imputed distributions, IVEware imputed higher values than PROC MI in the lower part of the distribution but not in the tail part (maximum value). To better visualize the difference in imputed values with lower counts, Figure 2 shows a scatter plot comparison of the counts imputed as less than 1,000 by both PROC MI (*count_mi1*) and IVEware (*count_ive1*). The main cluster of observation (red oval) sits around the 200 count marker for IVEware and just above 0 for PROC MI. The graph also shows a few higher counts (blue oval) imputed by PROC MI (above 500) that were imputed as counts below 500 for IVEware.

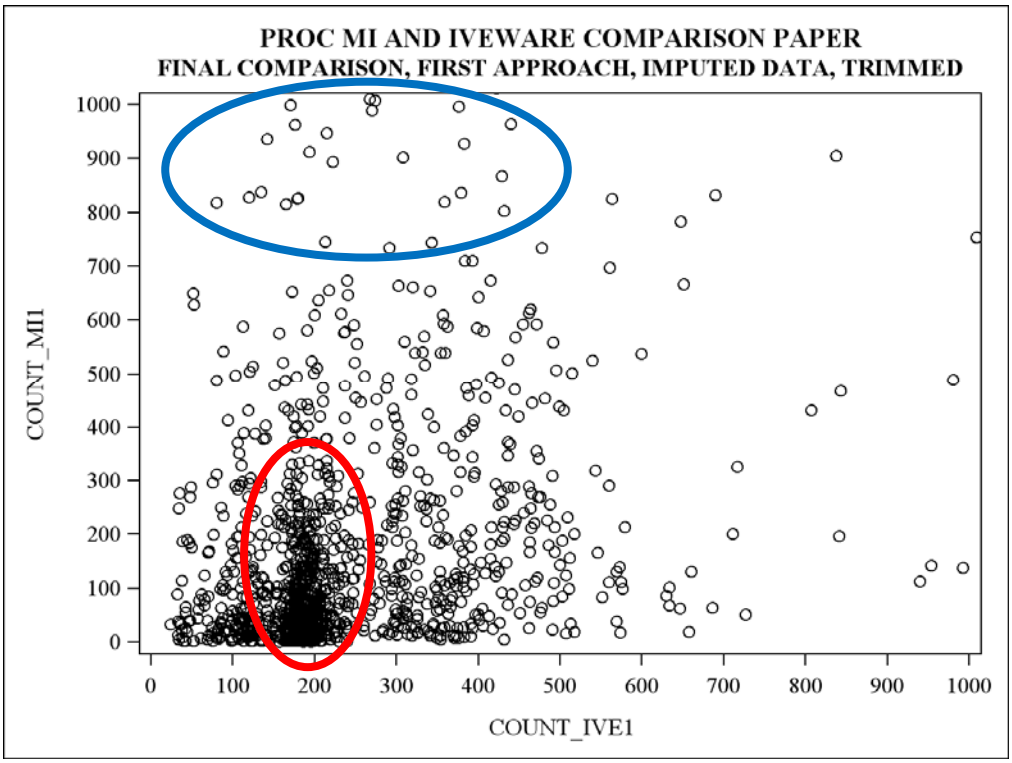

Figure 2. First Approach Scatter Plot of Both Methods

SECOND APPROACH

The results from the first approach show that neither imputation method addressed the heavy zero distribution accurately; therefore, a second approach was implemented. The second approach involves creating a flag to indicate whether the count is zero (flag = 0) or greater than zero (flag = 1), and then using this categorical flag to impute counts with zero or more. For comparison purposes, the response variables in the second approach were renamed to *count_mi2* and *count_ive2*.

To maintain consistency, we used the same imputation method in both PROC MI and IVEware to impute the newly created categorical flags. In PROC MI, we used the FCS approach with a logistic regression method using the same X_1–X_9 variables from the first approach as possible covariates. In IVEware, a logistic regression approach was also used with the same nine covariates. The categorical flag was only imputed once.

Table 4 shows the PROC MI and IVEware imputation results for the categorical flag, as both counts and percentages. We can see that PROC MI imputed the categorical flag 109 times as 0 (out of the 1,194 cases) compared to the 73 cases imputed as 0 by IVEware. Both categorical flag percentages are closer to the percentage in the observed data.

Table 4. Distribution of Imputation Flags

Imputed Value	Observed Data (Nonmissing)	%	PROC MI Imputation Flag (Imputed)	%	IVEware Imputation Flag (Imputed)	%
0	618	9.3	109	9.1	73	6.1
1	6,059	90.7	1,085	90.9	1,121	93.9
Total	6,677	100	1,194	100	1,194	100

For cases where the imputation flag was imputed as 0, the response variable in PROC MI (*count_mi2*) had to be manually edited to 0. PROC MI does not have a feature to restrict imputation to certain cases only—in this situation, cases where the categorical flag equals 1. In IVEware, we used the RESTRICT statement to only impute cases where the categorical flag equaled 1. IVEware will automatically impute a 0 value where the restriction is not met. Both the PROC MI and IVEware coding structures remained the same for the second step with the exception of the MINIMUM value and the RESTRICT statement in IVEware mentioned above. Given that the imputation flag was used to edit and impute the 0 counts, the second step had a minimum imputation value restriction of 1.

Table 5 shows the combined results for the second approach. The estimated mean for both methods in the second approach remained close to the estimate in the first approach, whereas the combined variance increased for PROC MI. On the other hand, IVEware shows no differences in the variance values, with a slight increase in the estimated mean. Comparing Table 3 with Table 5, we can see that the increase in PROC MI's total variance is due to a higher between-imputation variance value. The within-imputation variance value did not vary by much.

Table 5. Combined Results for Both PROC MI and IVEware on Second Approach

	Count_mi2	Count_ive2
Estimated mean	250.34	255.65
Between variance	1.73	0.01
Within variance	22.47	19.52
Combined variance	24.2	19.53

Again looking at the imputed distributions, in Figure 3, we can see counts imputed as fewer than 1,000 for the second approach, and note the same result for the cluster of imputed values (red oval) as in the first approach, with a slight shift to lower values for IVEware. More noticeable are the 0 imputed values for IVEware that are imputed as higher counts for PROC MI (green oval); in addition, as in the first approach, we can still see that a few missing values were imputed with higher counts by PROC MI than by IVEware (blue oval).

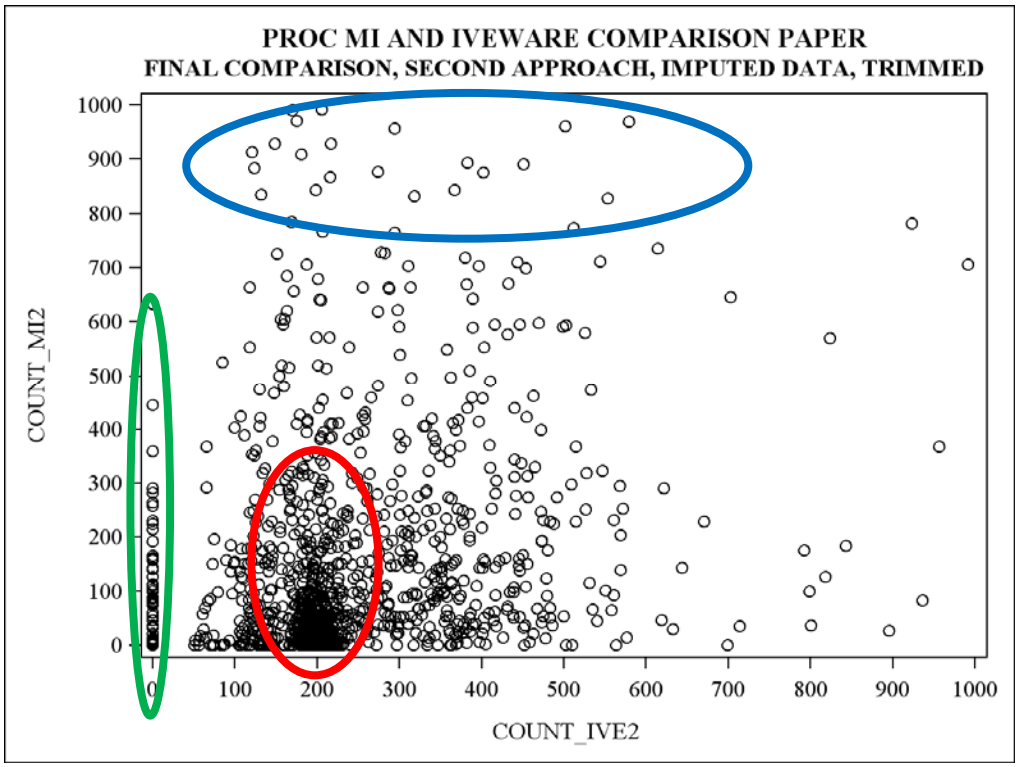

Figure 3. Second Approach Scatter Plot of Both Methods

SUMMARY AND CONCLUSION

The results in imputed values show the difference in methods used by PROC MI and IVEware. Procedure methodology differences from the two methods had an effect on the final results, where we can see the main difference in sequential imputation techniques. The first imputation approach, allowing a no-restriction imputation, did not address the heavy zero distribution, but showed higher maximum values imputed by PROC MI than by IVEware. In general, IVEware results were more conservative and distributed more closely together, whereas PROC MI had a more spread out range.

The second approach better addressed the heavy zero distribution issue; using the imputation flag, a percentage of zero values similar to the original percentage of zero values was imputed for both methods. The second step of this approach showed similar results for IVEware, whereas the variance for PROC MI increased. This is an indicator that, for our data example, the variance in the PROC MI procedure was overestimated more between data sets than within each data set.

A major aspect to point out that could have influenced the results is the transformation that was necessary for the PROC MI to better impute values. The ability of IVEware to model a variable of count nature using a Poisson regression allows the imputed values to have a similar distribution to the original. The need to make the appropriate transformation for the PROC MI approach will introduce error, since some distributions may be too complex to accomplish normally. IVEware allows the user to apply the appropriate distribution model to impute different types of variables. If IVEware is not available, PROC MI is a good method to impute variables, but careful transformation techniques maybe be required.

Selecting the most appropriate multiple imputation software for your particular data structure requires careful evaluation. The different approaches and methods available will result in different values being imputed and could affect your results. The nature of the variables to be imputed plays a big role in the method selection, as do the amount of data and the relationship between variables.

REFERENCES

Anscombe, F. J. (1948), The transformation of Poisson, binomial and negative-binomial data, *Biometrika, 35(3-4),* 246–254

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (Eds.) (1996). *Markov Chain Monte Carlo in Practice*. London: Chapman & Hall.

Little, R. J. A., and Rubin, D. B. (1987). *Statistical Analysis with Missing Data*. New York: J. Wiley & Sons.

Raghunathan, T. E., Lepkowski, J. M., Van Hoewyk, J., & Solenberger, P. (2001). A multivariate technique for multiply imputing missing values using a sequence of regression models.
Survey Methodology 27: 85–95.

Raghunathan, T. E., Solenberger, P. W., & Van Hoewyk, J. V. (2002). *IVEware: Imputation and Variance Estimation Software User's Guide*. Ann Arbor, MI: Institute for Social Research, University of Michigan. www.isr.umich.edurs/c/smp/ive/.

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

Rubin, D. B. (1996). Multiple Imputation After 18+ Years. *Journal of the American Statistical Association*, 91, 473–489.

Schafer, J. L. (1997). *Analysis of Incomplete Multivariate Data*, New York: Chapman and Hall.

Sterne, J. A., White, I. R., Carlin, J. B., Spratt, M., Royston, P., Kenward, M. G., Wood, A. M., & Carpenter, J. R. Multiple imputation for missing data in epidemiological and clinical research: potential and pitfalls. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2714692/.

Stuart, E. A. (2010). Recent advances in missing data methods: multiple imputation by chained equations. Presentation at AcademyHealth annual research meeting. Retrieved from http://www.academyhealth.org/files/2010/sunday/StuartE.pdf.

van Buuren S, Brand J. P. L., Groothuis-Oudshoorn, K., Rubin, D. B. (2006). Fully conditional specification in multivariate imputation. *Journal of Statistical Computation and Simulation*, *76*(12), 1049–1064.

White, I. R., Royston, P., and Wood, A. M. (2011). Multiple imputation using chained equations: Issues and guidance for practice. *Statistics in Medicine*, *30*, 377–399.

Yang, Yuan C. (2009). Multiple Imputation for Missing Data: Concepts and New Development (Version 9.0). Retrieved from http://www.math.montana.edu/~jimrc/classes/stat506/notes/multipleimputation-SAS.pdf.

Your comments and questions are valued and encouraged. Contact the authors at:
Bruno Vizcarra and Amang Sukasih, Mathematica Policy Research, 1100 1st Street NE, Washington, DC, 20002-4221. Phone 202-484-4231. Email bvizcarra@mathematica-mpr.com or asukasih@mathematica-mpr.com.