

A macro for nearest neighbor imputation

Lung-Chang Chien, University of North Carolina at Chapel Hill, Chapel Hill, NC
 Mark Weaver, Family Health International, Research Triangle Park, NC

ABSTRACT

SAS® software only has one data imputation procedure named PROC MI for multiple imputation to provide an easier way to impute missing data. However, it requires sufficient background to understand how to use it properly. In addition, multiple imputation needs strong assumption of data profile, but sometimes practical data cannot match it. This paper describes an alternative data imputation method named nearest neighbor imputation, and also provides an easy SAS macro along with a simulated data example and a case study.

INTRODUCTION

Missing data problem is a very common situation in practical studies. Most SAS procedures require the assumption that the missing data mechanism is missing completely at random (MCAR) or missing at random (MAR), which allows one to ignore those records with missing data (i.e., “casewise deletion”). However, this approach can be somewhat wasteful because we also eliminate other non-missing variables for any subject who only has missingness in a single variable. This is clearly inefficient if those non-missing variables are included in model fitting or statistical analysis. Moreover, some imputation methods require strong assumptions regarding the data distribution, biases result if those assumptions are violated. Therefore, one could consider applying a data imputation method with less restriction to make the whole data set more complete and use all available information for non-missing variables. The nearest neighbor imputation is a kind of hot deck imputation with a long history, and has been used in many surveys conducted by Statistics in Canada, the U.S. Bureau of Labor Statistics, and the U.S. Census Bureau. However, its statistical properties had not been derived until Chen and Shao (2001). As described by Little and Rubin (1987), all imputation methods also require the missing data to be at least MAR, if not MCAR; the same is clearly true for the nearest neighbor approaches described here. This paper provides a brief introduction to nearest neighbor imputation, provides an easy-to-use SAS macro for the method, and demonstrates its use in two examples using simulated data and case study, respectively.

The introduction explains the purpose and scope of your paper and provides readers with any general information they need to understand your paper.

METHODOLOGY

Suppose the data structure with m missing values for the row indices $i=n-m+1, \dots, n$ can be re-expressed by

$$\begin{array}{c} \text{observed} \\ \left\{ \begin{array}{l} X_1 \\ \cdot \\ \cdot \\ X_{n-m} \\ X_{n-m+1} \end{array} \right. \\ \text{missing} \\ \left\{ \begin{array}{l} \cdot \\ \cdot \\ \cdot \\ X_n \end{array} \right. \end{array} \quad \text{observed} \quad \left\{ \begin{array}{l} Y_1 \\ \cdot \\ \cdot \\ Y_{n-m} \\ Y_{n-m+1} \\ \cdot \\ \cdot \\ Y_n \end{array} \right.$$

A missing value X_j , $j=n-m+1, \dots, n$, is imputed by choosing that value X_i , $i=1, \dots, n-m$, which is corresponding to its closest value Y_j to Y_i . This is also the true meaning of *nearest neighbor*. The definition of *closest* is determined by the distance between any two response values. In other words, the distance of the nearest neighborhood is calculated with all observed values for Y by

$$|Y_i - Y_j| = \min_{1 \leq k \leq n-m} |Y_k - Y_j|$$

When we find the response value Y_k , $k=1, \dots, n-m$, which is the closest one to Y_j , $j=1, \dots, n$, then we can impute its corresponding X_k to missing value

If there are more than one X_k whose corresponding response values Y_k has the same minimum distance to Y_j among others, then the mean of those X -values is imputed.

SIMULATION DATA

In this paper, a simulated data set will be used to apply a macro program with nearest neighbor imputation. In this simulated data set, there are two variables named X and Y , respectively. In the beginning, both variables are all complete, and the sample size is 200. X is generated from uniform distribution $U(0,1)$, and Y is generated from the following polynomial equation:

$$Y_i = 0.2 + 0.5X_i - 0.7X_i^2 + 0.4X_i^3 + \varepsilon_i, \quad i = 1, \dots, 200; \quad \varepsilon_i \sim N(0, 0.1)$$

Then, a random sampling can be applied to determine missing data randomly. The following process can generate an incomplete data set with missing values in X which mechanism follows MCAR.

```
PROC SURVEYSELECT DATA=SIMDATA OUT=SIMDATA_SRS METHOD=SRS SAMPSIZE=190 SEED=1234;
RUN;
DATA SIMDATA_SRS;
  SET SIMDATA_SRS;
  NOMISS=1;
RUN;

DATA SIMDATA_MISS;
  MERGE SIMDATA SIMDATA_SRS;
  BY ID;
  IF NOMISS=. THEN X=.;
  DROP NOMISS SELECTIONPROB SAMPLINGWEIGHT;
RUN;
```

The missing rate in this simulated data set is defined as 5%, so 5 X -values are missing by the above program.

MACRO

A macro named %NNI will be introduced in this section. The call for the macro %NNI is as follows:

```
%NNI (INDATA=,
      MISSVAR=,
      RESPVAR=,
      IDVAR=,
      OUTDATA=) ;
```

There are five arguments in this macro.

1. INDATA: the name of input data set, including library name.
2. MISSVAR: scalar variable that contains some missing values to be imputed.
3. RESPVAR: response variable. If there is no specific response variable, any variable which is other than the variable in MISSVAR is allowable. If there are more than two choices, the variable with the highest correlation coefficient with the variable in MISSVAR is the best choice.
4. IDVAR: variable name of subject ID. It is recommended to include this variable in data set if there is no ID-like variable.
5. OUTDATA: name final data set with imputed data.

The following code is the application of using %NNI in previous simulation data.

```
%NNI (INDATA=SIM.SIMDATA_MISS,
      MISSVAR=X,
      RESPVAR=Y,
      IDVAR=ID,
      OUTDATA=SIM.SIMDATA_NNI) ;
```

The scatter plot in figure 1 shows that imputed data tends to be very close to the unobserved missing value based on observed X and Y . The demographics of X also show the profile is similar to X 's true descriptive statistics after using nearest neighbor imputation.

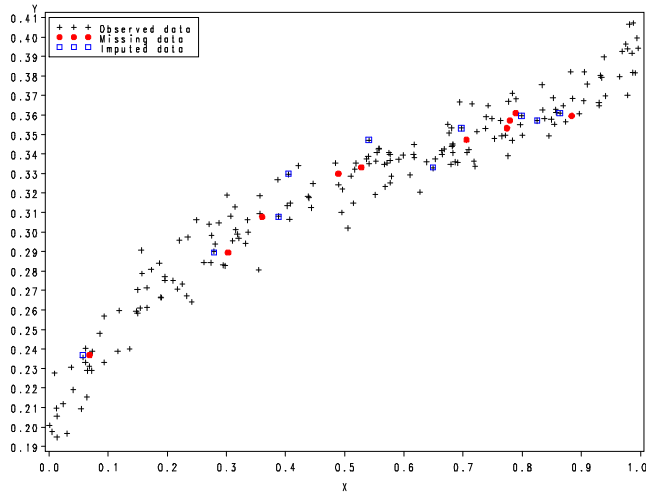


Figure 1. The scatter plot of Y with observed, missing and imputed values of X

Summary statistics for X

	<i>True value</i>	<i>Before imputation</i>	<i>After imputation</i>
Mean	0.5181	0.5155	0.5172
STD	0.2942	0.2961	0.2941
Min	0.0009	0.0009	0.0009
Q1	0.2721	0.2622	0.2721
Median	0.5537	0.5537	0.5537
Q3	0.7686	0.7637	0.7646
Max	0.9967	0.9967	0.9967

CASE STUDY

A case study about daily temperature and centralized particulate matter $< 2.5 \mu\text{m}$ in aerodynamic diameter ($\text{PM}_{2.5}$) in Raleigh is applied here. The source of the data set is the National Morbidity Mortality Air Pollution Study (NMMAPS) which is maintained by the Department of Biostatistics in Johns Hopkins University. The period of data is from 1987 to 2000, but only 2000's data are used here. Each variable has 366 observations. Daily temperature data are complete, but $\text{PM}_{2.5}$ has 17 missing values. The missing rate is 4.64%.

The following codes are used to impute missing data:

```
%NNI (INDATA=CASE.RALFORNNI,
      MISSVAR=PM25TMEAN,
      RESPVAR=TMPD,
      IDVAR=DATE,
      OUTDATA=CASE.RAL_NNI);
```

All imputed data are generated by known information of daily temperature, and plotted by red dots in figure 2. The demographics for $\text{PM}_{2.5}$ before-imputation and after-imputation are very similar. The correlation between $\text{PM}_{2.5}$ and daily temperature before imputation is 0.3113, and it's 0.3136 after imputation. Suppose this data set is used to be fit by generalized additive model (GAM), we can show how influence of imputing data by the nearest neighbor imputation in model fitting.

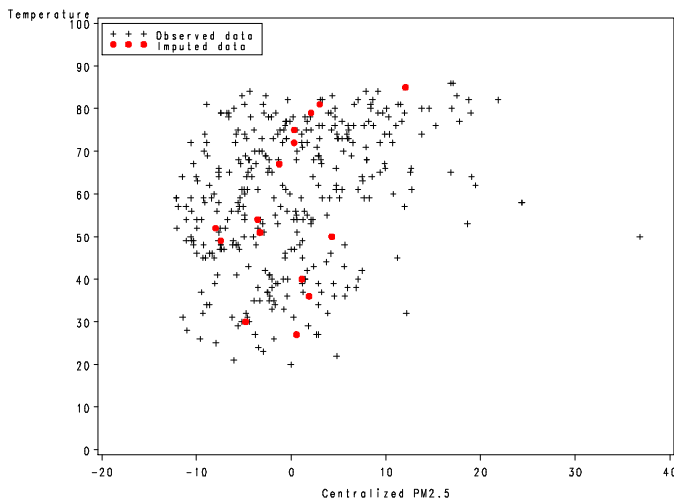


Figure 2. The scatter plot of daily temperature with observed and imputed centralized $\text{PM}_{2.5}$.

Summary statistics for $\text{PM}_{2.5}$

	<i>Before imputation</i>	<i>After imputation</i>
Mean	0.2297	0.2148
STD	7.6452	7.5347
Min	-12.1366	-12.1366
Q1	-5.3750	-5.2193
Median	-0.6531	-0.6257
Q3	4.7307	4.6272
Max	36.8448	36.8448

Before imputation					After imputation				
Regression Model Analysis Parameter Estimates					Regression Model Analysis Parameter Estimates				
Parameter	Parameter Estimate	Standard Error	t Value	Pr > t	Parameter	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	58.27474	0.76187	76.49	<.0001	Intercept	57.70478	0.73717	78.28	<.0001
pm25tmean	0.33609	0.04986	6.74	<.0001	pm25tmean	0.33837	0.04900	6.91	<.0001
Linear(t)	0.00602	0.00362	1.66	0.0974	Linear(t)	0.00821	0.00349	2.35	0.0192
Smoothing Model Analysis Analysis of Deviance					Smoothing Model Analysis Analysis of Deviance				
Source	DF	Sum of Squares	Chi-Square	Pr > ChiSq	Source	DF	Sum of Squares	Chi-Square	Pr > ChiSq
Spline(t)	6.0000	66097	1321.3932	<.0001	Spline(t)	6.0000	70078	1422.1678	<.0001

Table 1. Outputs of PROC GAM before imputation and after imputation

The model is defined as:

$$(\text{TEMPERATURE})_t = \beta_0 + \beta_1 \times (\text{PM}_{2.5})_t + f(\text{Time}, \text{df} = 7) + \varepsilon_t \quad t = 1, \dots, 366$$

The results of model fitting are shown in table 1. The estimated coefficient of PM_{2.5} after imputation is a little bit higher than that before imputation; however, its standard error is reduced with imputed data. The estimated smoothing function of time with imputed PM_{2.5} data is also similar as that without imputed PM_{2.5} data (figure 3).

LIMITATION

In original methodology of nearest neighbor imputation, there is no special restriction. Both continuous and categorical variable can apply it. However, this macro is not suitable for imputed missing categorical data because if there are more than one closest observed response for a missing datum, it is likely to generate imputed values with decimals. A modified approach is using an additional data step to round imputed data into integers. In addition, even though there is no study to support how large data set it can support, but too small sample size may cause problems. A trial reminder is that there should be at least one variable that is totally complete without any missing value. Even though this complete variable is not a response variable in analysis, you can assume it is. If there are more than one complete variable, you can easily use PROC CORR procedure to make a correlation matrix among them, and pick the complete variables with the highest correlation with missing variable.

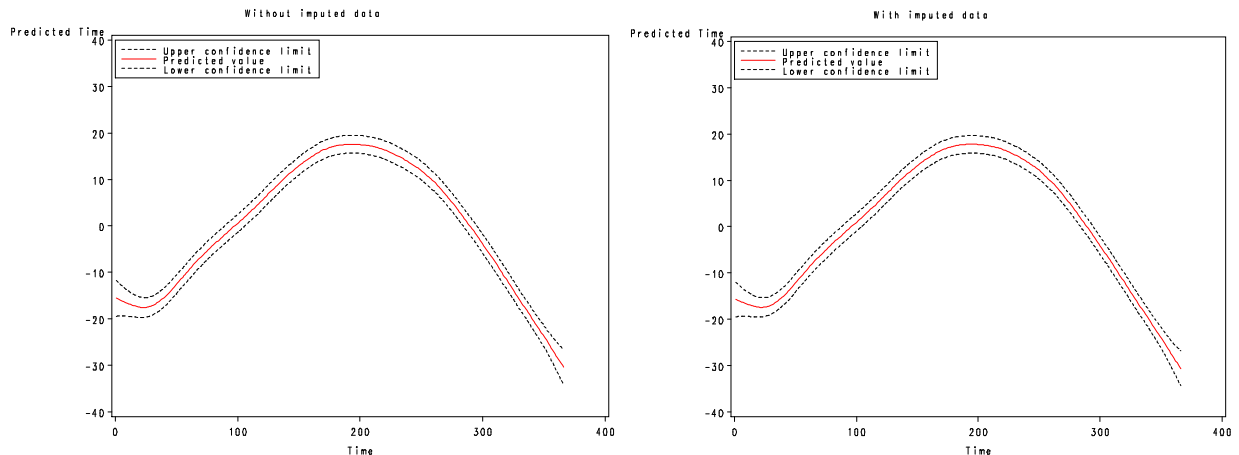


Figure 3. Estimated smoothing functions with 95% confidence interval before and after using imputation.

REFERENCES

- Chen, J., and Shao, J. (2000). "Nearest neighbor imputation for survey data," *Journal of Official Statistics*, Vol.16, No.2, 113-131.
- Little, R. J. A., and Rubin, D. B. (1987). *Statistical Analysis with Missing Data*. New York: John Wiley & Sons.
- Nittner, T. (2002). "The additive model with missing values in the independent variable – theory and simulation," Collaborative Research Center 386, Discussion Paper 272.
- Van Hulse, J., and Khoshgoftaar T.M. (2007). "Incomplete-case nearest neighbor imputation in software measurement data," Proceedings of the IEEE International Conference on Information Reused Integration (IRI 2007), 630-637.
- SAS Institute Inc., SAS Macro Language: Reference, Version 8, Cary, NC: SAS Institute, Inc., 1999.

CONTACT INFORMATION

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

Lung-Chang Chien
University of North Carolina at Chapel Hill
Chapel Hill, NC 27599
E-mail: cchien@email.unc.edu

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.
Other brand and product names are trademarks of their respective companies.

APPENDIX

```
%MACRO NNI (INDATA=,                                /*INPUT DATA SET (INCLUDING LIBRARY NAME) */
             MISSVAR=,                                /*INPUT SINGLE VARIABLE WITH MISSING DATA*/
             RESPVAR=,                                /*RESPONSE VARIABLE*/
             IDVAR=,                                  /*SUBJECT VARIABLE*/
             OUTDATA=                                /*OUTPUT DATA SET WITH COMPLETE DATA*/
             );

DATA OBSY_MISSX;
  SET &INDATA;
  IF &MISSVAR=.;
  KEEP &IDVAR &RESPVAR;
RUN;

PROC MEANS DATA=&INDATA N NMIS NOPRINT;
  VAR &MISSVAR;
  OUTPUT OUT=OBSN N=OBSX NMIS=MISSX;
RUN;

DATA _NULL_;
  SET OBSN;
  CALL SYMPUT ('OBSN', OBSX);
  CALL SYMPUT ('MISSN', MISSX);
RUN;

DATA SIMDATA_NOMISS;
  SET &INDATA;
  IF &MISSVAR NE .;
RUN;

PROC IML;
  USE SIMDATA_NOMISS;
  READ ALL VAR {&MISSVAR} INTO XMAT;
  READ ALL VAR {&RESPVAR} INTO YMAT;
```

```

USE OBSY_MISSX;
READ ALL VAR {&IDVAR &RESPVAR} INTO MISSMAT;
TXMAT=T(XMAT);
TYMAT=T(YMAT);
DISTANCE=J(&MISSN,&OBSN,.);
MIND=J(&MISSN,1,.);
MISSVAR=J(&MISSN,&OBSN,.);
IMP=J(&MISSN,1,.);
DO I = 1 TO &MISSN;
    DO J = 1 TO &OBSN;
        DISTANCE[I,J]=ABS(MISSMAT[I,2]-TYMAT[,J]);
    END;
    MIND[I,]=MIN(DISTANCE[I,]);
END;
DO I = 1 TO &MISSN;
    DO J = 1 TO &OBSN;
        IF DISTANCE[I,J]=MIND[I,] THEN MISSVAR[I,J]=TXMAT[,J];
    END;
    IMP[I,]=MISSVAR[I,:];
END;
CNAME={"&IDVAR" "&RESPVAR" "&MISSVAR"};
IMPX=MISSMAT||IMP;
CREATE NNI FROM IMPX[C=CNAME];
APPEND FROM IMPX;
QUIT;

DATA &OUTDATA;
    MERGE &INDATA NNI;
    BY &IDVAR;
RUN;
%MEND;

```