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 fro 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,…,n can be re-expressed by

\[
\begin{align*}
\text{observed} & : \begin{cases} X_1 \\ \vdots \\ X_{n-m} \\ X_{n-m+1} \end{cases} & : \begin{cases} Y_1 \\ \vdots \\ Y_{n-m} \\ Y_{n-m+1} \end{cases} \\
\text{missing} & : X_n \\
\end{align*}
\]

A missing value \(X_j, j=n-m+1,…,n\), is imputed by choosing that value \(X_i, i=1,…,n-m\), which is corresponding to its closest value \(Y_i\) to \(Y_j\). 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 \(Y_k, k=1,…,n-m\), which is the closest one to \(Y_j, j=1,…,n\), 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 \( Y_i \) 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:

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

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.
CASE STUDY
A case study about daily temperature and centralized particulate matter < 2.5 μm in aerodynamic diameter (PM2.5) 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 PM2.5 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 PM2.5 before-imputation and after-imputation are very similar. The correlation between PM2.5 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.

Summary statistics for X

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

Summary statistics for PM2.5

<table>
<thead>
<tr>
<th></th>
<th>Before imputation</th>
<th>After imputation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.2297</td>
<td>0.2148</td>
</tr>
<tr>
<td>STD</td>
<td>7.6452</td>
<td>7.5347</td>
</tr>
<tr>
<td>Min</td>
<td>-12.1366</td>
<td>-12.1366</td>
</tr>
<tr>
<td>Q1</td>
<td>-5.3750</td>
<td>-5.2193</td>
</tr>
<tr>
<td>Median</td>
<td>-0.6531</td>
<td>-0.6257</td>
</tr>
<tr>
<td>Q3</td>
<td>4.7307</td>
<td>4.6272</td>
</tr>
<tr>
<td>Max</td>
<td>36.8448</td>
<td>36.8448</td>
</tr>
</tbody>
</table>

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

Figure 2. The scatter plot of daily temperature with observed and imputed centralized PM2.5.
The model is defined as:

\[(\text{TEMPERATURE})_t = \beta_0 + \beta_1 \times (\text{PM}_{2.5})_t + f(\text{Time}, df = 7) + \epsilon_t \quad t = 1, ..., 366\]

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.](image-url)
REFERENCES

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

```sas
%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 NMISS NOPRINT;
  VAR &MISSVAR;
  OUTPUT OUT=OBSN N=OBSX NMISS=MISSX;
RUN;

DATA _NULL_;            /*INPUT DATA SET WITH COMPLETE DATA*/
  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;