Balanced imputation for Swiss cheese nonresponse

Audrey-Anne Vallée, Esther Eustache and Yves Tillé

*University of Neuchatel*

The Use of R in Official Statistics

December 2, 2020
Overview

Introduction

Introduction to the nonresponse

Notations of Swiss cheese nonresponse

Imputation method

Requirements of the imputation method

Imputation probabilities

Imputation matrix

Example with the R software
The nonresponse

Individual for whom we want to observe $J$ variables to obtain vector $\mathbf{x}_k = (x_{1k}, x_{2k}, \ldots, x_{jk})^T$. 
The nonresponse

\[ S(n) \]

**Respondent:**
\[ \mathbf{x}_k = (x_{1k}, x_{2k}, \ldots, x_{Jk})^T \]
is fully observed.

**Non-respondent:**
\[ \mathbf{x}_k = (x_{1k}, x_{2k}, \ldots, x_{Jk})^T \]
contains at least one missing or not usable value.
The **Swiss cheese nonresponse** is:

- not monotone (i.e. without any pattern).
- contained in several or all variables.
The **Swiss cheese nonresponse** is:

- not monotone (i.e. without any pattern).
- contained in several or all variables.

**Idea:** generalize the method for the univariate nonresponse of Hasler and Tillé (2016) to the multivariate one.
The **Swiss cheese nonresponse** is:

- not monotone (i.e. without any pattern).
- contained in several or all variables.

**Idea:** generalize the method for the univariate nonresponse of Hasler and Tillé (2016) to the multivariate one.

![Swiss cheese](image)

**Figure 2:** The univariate case: a slice of Swiss cheese with holes.
The **Swiss cheese nonresponse** is:

- not monotone (i.e. without any pattern).
- contained in several or all variables.

*Idea:* generalize the method for the univariate nonresponse of Hasler and Tillé (2016) to the multivariate one.

---

**Figure 2:** The univariate case: a slice of Swiss cheese with holes.

**Figure 3:** The multivariate case: a Swiss cheese with holes.
- Population $U$ of size $N$.
- $J$ variables of interest.
- Sample $S \subseteq U$ of size $n$.
- For each unit $k \in S$: $\mathbf{x}_k = (x_{k1}, \ldots, x_{kj}, \ldots, x_{kJ})^\top$. 
Notations of Swiss cheese nonresponse

- Population $U$ of size $N$.
- $J$ variables of interest.
- Sample $S \subset U$ of size $n$.
- For each unit $k \in S$: $x_k = (x_{k1}, \ldots, x_{kj}, \ldots, x_{kJ})^\top$.

- Sample $S_r \subset S$ of size $n_r$ contains completely observed units.
- Sample $S_m \subset S$ of size $n_m$ contains units with at least one missing or not usable value.
- $S_r \cup S_m = S$ and $n_r + n_m = n$.
- Not monotone nonresponse.
Properties required for an imputation method:

- Impute by realistic values.
- Preserve the distribution of the variables.
- Preserve the relationships between the variables.
Requirements of the imputation method

Properties required for an imputation method:

- Impute by realistic values.
- Preserve the distribution of the variables.
- Preserve the relationships between the variables.

Requirements of our method:

(i) Donor imputation method: choose one donor for each nonrespondent in $S_m$ among units in $S_r$ to impute its missing values.

\[ x_k = (A, 4, 10) \quad \text{donor} \quad \Rightarrow \quad x_k' = (A, n, 10) \]
Properties required for an imputation method:

- Impute by realistic values. ✓
- Preserve the distribution of the variables.
- Preserve the relationships between the variables. ✓

Requirements of our method:

(i) Donor imputation method: choose one donor for each nonrespondent in $S_m$ among units in $S_r$ to impute its missing values.

(ii) Each donor should be selected among the $K$ nearest respondents of each nonrespondent unit.
Requirements of the imputation method

Properties required for an imputation method:

- Impute by realistic values. ✓
- Preserve the distribution of the variables. ✓
- Preserve the relationships between the variables. ✓

Requirements of our method:

(i) Donor imputation method: choose one donor for each nonrespondent in $S_m$ among units in $S_r$ to impute its missing values.

(ii) Each donor should be selected among the $K$ nearest respondents of each nonrespondent unit.

(iii) If the observed values of the nonrespondents were imputed, the total estimator of each variable should remain unchanged.
- Let $\psi = (\psi_{uv})$ denote the matrix of size $n_r \times n_m$ containing imputation probabilities, where $(u, v) \in S_r \times S_m$.

- $\psi_{uv}$: probability that the respondent $u \in S_r$ gives its values to the nonrespondent $v \in S_m$.

$$
\psi = 
\begin{pmatrix}
\psi_{11} & \psi_{12} & \psi_{13} \\
\psi_{21} & \psi_{22} & \psi_{23} \\
\psi_{31} & \psi_{32} & \psi_{33} \\
\psi_{41} & \psi_{42} & \psi_{43} \\
1 & 1 & 1
\end{pmatrix}
$$
Constraints on $\psi$: 


Constraints on $\psi$:

1. Each nonrespondent must be imputed by only one respondent:

$$\sum_{i \in s_r} \psi_{uv} = 1.$$
Constraints on $\psi$:

1. Each nonrespondent must be imputed by only one respondent:

$$\sum_{i \in s_r} \psi_{uv} = 1.$$  

2. Each donor selected among the $K$ nearest neighbors of each nonrespondent unit:

$$\psi_{uv} = 0 \text{ if } u \notin \text{knn}(v)$$

where $\text{knn}(\ell) = \{u \in s_r \mid \text{rank}(d(u, v)) \leq K\}$ and $d(., .)$ is a distance function.
Constraints on $\psi$:

1. Each nonrespondent must be imputed by only one respondent:
   \[ \sum_{i \in s_r} \psi_{uv} = 1. \]

2. Each donor selected among the $K$ nearest neighbors of each nonrespondent unit:
   \[ \psi_{uv} = 0 \text{ if } u \notin \text{knn}(v) \]
   where $\text{knn}(\ell) = \{ u \in s_r \mid \text{rank}(d(u, v)) \leq K \}$ and $d(., .)$ is a distance function.

3. If the observed values of the nonrespondents were imputed, the total estimator of each variable should remain unchanged:
   \[ \sum_{v \in S_m} r_{vj} \sum_{u \in S_r} \psi_{uv} x_{uj} = \sum_{v \in S_m} r_{vj} x_{vj}, \]
   where $r_{vj}$ is 1 if unit $v$ responded to variable $j$ and 0 otherwise.
Steps to obtain final matrix $\psi$:

Step 1. Initialization of $\psi$:

$$
\psi_{uv} = \begin{cases} 
\frac{1}{k} & \text{if } u \in \text{knn}(v), \\
0 & \text{otherwise.}
\end{cases}
$$
Imputation probabilities

Steps to obtain final matrix $\psi$:

Step 1. Initialization of $\psi$:

$$\psi_{uv} = \begin{cases} 
\frac{1}{K} & \text{if } u \in \text{knn}(v), \\
0 & \text{otherwise}.
\end{cases}$$

Step 2. Update $\psi$ using an algorithm of calibration proposed by Deville and Särndal (1992) in order to satisfy requirements 1-3.
Matrix of imputation probabilities: $\psi = \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0 \\ 0.3 & 0 & 0.4 \\ 0.2 & 0 & 0.1 \end{pmatrix}$

Imputation matrix: $\phi = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

Element $\phi_{uv} = 1$ means that missing values of nonrespondent $v$ will be imputed by values of respondent $u$:

$$x^*_u = \sum_{v \in S_r} \phi_{uv} x_{v}.$$
Matrix of imputation probabilities:

\[ \psi = \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0 \\ 0.3 & 0 & 0.4 \\ 0.2 & 0 & 0.1 \end{pmatrix} \]

\[ \rightarrow \]

Imputation matrix:

\[ \phi = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \]

Element \( \phi_{uv} = 1 \) means that missing values of nonrespondent \( v \) will be imputed by values of respondent \( u \):

\[ x_{uj}^* = \sum_{v \in S_r} \phi_{uv} x_{vj}. \]

Step to obtain final matrix \( \phi \):

**Step 3.** Compute \( \phi \) using a stratified sampling method such that:

- Columns of \( \psi \) correspond to stratum.
- Balancing constraints of requirement 3. is satisfied.
```r
# Download the package SwissCheese
# library(devtools)
# install_github("EstherEustache/SwissCheese@master")
# library(SwissCheese)

# Dataframe with NA values
Sm <- as.vector(attr(stats::na.omit(X_NA),"na.action"))
Sm

## [1] 18 21 29 5 1 10 17 20

Sr <- which(!(1:nrow(X) %in% Sm))
Sr

## [1]  2  3  4  6  7  8  9 11 12 13 14 15 16 19 22 23 24 25 26 27 28
```
# Nonrespondents

```r
define
    head(X_NA[Sm,])

##
##   V1    V2    V3
## [1,] NA 49.48603 1
## [2,] NA 36.05060 0
## [3,] NA 19.34894 0
## [4,] 21.12337 NA 0
## [5,] 36.64376 47.15358 NA
## [6,] 23.75826 33.93555 NA
```

# Respondents

```r
define
    head(X_NA[Sr,])

##
##   V1    V2    V3
## [1,] 47.19283 57.77238 1
## [2,] 42.91603 56.86644 1
## [3,] 57.71289 77.52506 1
## [4,] 40.32247 53.35982 1
## [5,] 52.91569 64.01816 1
## [6,] 63.35500 67.27140 1
```
## Swiss cheese imputation

```r
SW <- swissCheeseImput(X = X NA, d = NULL, k = NULL, 
                        tol = 1e-3, max_iter = 50)
```

###---Optimal number of neighbors considered

```r
SW$k
```

### [1] 4

###---The nonrespondent imputed

```r
head(SW$X_new[Sm,])
```

###

<table>
<thead>
<tr>
<th></th>
<th>V1</th>
<th>V2</th>
<th>V3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>36.93283</td>
<td>49.48603</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>29.43678</td>
<td>36.05060</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>17.39018</td>
<td>19.34894</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>21.12337</td>
<td>45.00952</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>36.64376</td>
<td>47.15358</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>23.75826</td>
<td>33.93555</td>
<td>0</td>
</tr>
</tbody>
</table>