VERSITA

Journal of Official Statistics, Vol. 29, No. 2, 2013, pp. 299–321, DOI: 10.2478/jos-2013-0024

Calibrated Hot-Deck Donor Imputation Subject to Edit Restrictions

Wieger Coutinho[1], *Ton de Waal*[2], *and Natalie Shlomo*[3]

A major challenge faced by basically all institutes that collect statistical data on persons, households or enterprises is that data may be missing in the observed data sets. The most common solution for handling missing data is imputation. Imputation is complicated owing to the existence of constraints in the form of edit restrictions that have to be satisfied by the data. Examples of such edit restrictions are that someone who is less than 16 years old cannot be married in the Netherlands, and that someone whose marital status is unmarried cannot be the spouse of the head of household. Records that do not satisfy these edits are inconsistent, and are hence considered incorrect. A further complication when imputing categorical data is that the frequencies of certain categories are sometimes known from other sources or have previously been estimated. In this article we develop imputation methods for imputing missing values in categorical data that take both the edit restrictions and known frequencies into account.

Key words: Categorical data; edit rules; imputation; population frequencies.

1. Introduction

National statistical institutes (NSIs) publish figures on many aspects of society. To this end, NSIs collect and process data on persons, households, enterprises, public bodies, and so on. A major challenge faced by NSIs is that data may be missing from the collected data sets. Some units that are selected for data collection cannot be contacted or may refuse to respond altogether. This is called unit nonresponse. For many individual units, data on some of the items may be missing. Persons may, for instance, refuse to provide information on their income or on their sexual habits, while at the same time giving answers to other, less sensitive questions on the questionnaire. Enterprises may not provide answers to certain questions, because they may consider it too complicated or too time consuming to answer these specific questions. Missing items of otherwise responding units is called item nonresponse. Whenever we refer to missing data in this article we will mean item nonresponse, rather than unit nonresponse.

In the statistical literature, ample attention is paid to missing data. The most common solution for handling missing data in data sets is imputation, where missing values are

[1] Loket Aangepast-Lezen, PO Box 84010, 2508 AA The Hague, The Netherlands
[2] Statistics Netherlands, PO Box 24500, 2490 HA The Hague, The Netherlands Emails: t.dewaal@cbs.nl and twal@cbs.nl
[3] School of Social Sciences, University of Manchester, Humanities Bridgeford Street, Manchester, M13 9PL, United Kingdom Email: natalie.shlomo@manchester.ac.uk

estimated and filled in. An important problem of imputation is to preserve the statistical distribution of the data set. This is a complicated problem, especially for high-dimensional data. For more on this aspect of imputation and on imputation in general we refer to several articles and books on imputation, such as Kalton and Kasprzyk (1986), Rubin (1987), Schafer (1997), Little and Rubin (2002), Longford (2005), and De Waal et al. (2011). Imputation methods can be divided into two broad classes: methods for categorical data and methods for numerical data. In the present article we focus on imputation of missing categorical data.

At NSIs the imputation problem is further complicated owing to the existence of constraints in the form of edit restrictions, or edits for short, that have to be satisfied by the data. Examples of such edits are that someone who is less than 16 years old cannot be married in the Netherlands, and that someone whose marital status is unmarried cannot be the spouse of the head of household. Records that do not satisfy these edits are inconsistent, and are hence considered incorrect. The problem of missing categorical data having to satisfy edits is examined by Winkler (2003) and De Waal et al. (2011).

A further complication for categorical data is that the frequencies of certain categories are sometimes known from other sources or have previously been estimated. Such frequencies will also be referred to as totals in this article. A population frequency of a category may, for instance, be known from an available related register. Alternatively, previously estimated frequencies may be known, and assumed fixed. In the Dutch Social Statistical Database estimated frequencies are fixed and later used to calibrate estimates of other quantities (see Houbiers 2004, and Knottnerus and Van Duin 2006). In fact, this strategy of fixing frequencies and later using these fixed frequencies to calibrate other quantities to be estimated forms the basis of the so-called repeated weighting method: a weighting method designed to obtain unified estimates when combining data from different sources.

In the present article we develop imputation methods for categorical data that take edits and known frequencies into account. The problem of imputation of missing categorical data having to satisfy edits and to preserve totals is also discussed in Favre et al. (2005). In contrast to the methods proposed here, the imputation is not used as an estimation technique, rather as a way to obtain consistency with edits and previously estimated totals. Another difference is that in Favre et al. (2005) only one variable to be imputed is considered. Their method does not guarantee that edits involving several variables to be imputed will be satisfied. The related problem of imputation of missing numerical data having to satisfy edits and to preserve totals is examined in Pannekoek et al. (2008). Liu and Rancourt (1999) discuss imputation of missing categorical data having to preserve totals. They do not consider edits, however.

The imputation methods developed in this article are intended to be used in the situation where one wants to impute all units of the (sub-)population under consideration. By imputing, we pursue three goals:

- To preserve the statistical distribution of the true, but unknown, data as well as possible.
- To facilitate further processing, for example producing statistical tables after imputation is simply a matter of counting, without having to worry about inconsistencies between various tables or logical inconsistencies.

- To integrate data from different sources; for example, microdata from one source are calibrated to totals from another source. In this sense the imputation methods we propose in this article can be seen as data integration techniques (also see ESSnet on Data Integration 2011).

A word of caution is in place here: an imputed data set should not be seen as a restored complete data set. In particular, in an imputed data set variances may be underestimated and correlations may be disturbed, despite attempts to preserve them as well as possible. Estimating variances and correlations taking into account the imputations is quite complex and will not be considered in this article. For an overview of methods for estimating variances with data that have undergone imputations, we refer to Chapter 10 by Haziza in Pfefferman and Rao (2009).

Rubin (1976) introduced a classification of missing data mechanisms. He distinguishes between Missing Completely At Random (MCAR), Missing At Random (MAR) and Not Missing At Random (NMAR). Roughly speaking, in the case of MCAR there is no relation between the missing data pattern, that is, which data are missing, and the values of the data, either observed or missing. In the case of MAR there is a relation between the missing data pattern and the values of the observed data, but not between the missing data pattern and the values of the missing data. Using the values of the observed data one can then correct for the relation between the missing data pattern and the values of the observed data, since within classes of the observed data the missing data mechanism is MCAR again. In the case of NMAR there is a relation between the missing data pattern and the values of the missing data. Such a relation cannot be corrected for without positing a model. Given that the missing data mechanism is either MCAR or MAR, we can test whether the data are MCAR or MAR. However, there are no statistical tests to differentiate between MCAR/MAR and NMAR. In practice, the only way to distinguish MCAR/MAR from NMAR is by logical reasoning. For more on missing data mechanisms we refer to Little and Rubin (2002), McKnight et al. (2007) and Schafer (1997).

In this article we assume that the missing data mechanism is MCAR. Our imputation methods can, however, easily be extended to the case of MAR, by constructing imputation classes within which the missing data mechanism is MCAR.

The remainder of this article is organized as follows. Section 2 introduces the edit restrictions we consider in this article. Section 3 describes the imputation algorithms we have developed for our problem. An evaluation study using real data is described in Section 4. Finally, Section 5 ends the article with a brief discussion.

2. Edits and Frequencies for Categorical Data

2.1. Edits for Categorical Data

We denote the number of variables by n. Furthermore, we denote the domain, that is the set of all allowed values of a variable i, by Dom_i. All domains are assumed to be non-empty. In the case of categorical data, an edit j is usually written in so-called *normal form*, that is

as a Cartesian product of non-empty sets $F_i^j (i = 1, 2, \ldots, n)$:

$$F_1^j \times F_2^j \times \ldots \times F_n^j,$$

meaning that if for a record with values $(v_1, v_2, \ldots, v_n)$ we have $v_i \in F_i^j$ for all $i = 1, 2, \ldots, n$, then the record fails edit j, otherwise the record satisfies edit j. One generally demands that at least one of the $F_i^j (i = 1, 2, \ldots, n)$ should be a proper subset of the domain Dom_i, that is, should be strictly contained in Dom_i, as the "edit" with all $F_i^j (i = 1, 2, \ldots, n)$ equal to Dom_i cannot be failed by any record.

Example: Suppose we have three variables: *Marital Status*, *Age* and *Relation to Head of Household*. The possible values *of Marital Status* are "Married", "Unmarried", "Divorced" and "Widowed", of *Age* "<16 years" and "≥ 16 years", and of *Relation to Head of Household* "Spouse", "Child", and "Other". Suppose we have two edits, the first edit saying that someone who is less than 16 years cannot be married, and the second one that someone who is not married cannot be the spouse of the head of household. In normal form the first edit can be written as

$$(\{\text{Married}\}, \{< 16\,\text{years}\}, \{\text{Spouse, Child, Other}\}), \tag{1}$$

and the second one as

$$(\{\text{Unmarried, Divorced, Widowed}\}, \{< 16\,\text{years}, \geq 16\,\text{years}\}, \{\text{Spouse}\}). \tag{2}$$

2.2. *Frequencies for Categorical Data*

When a frequency for categorical data is known, for instance because it has already been estimated in another source, this simply means that one knows how many units in the data set should have a specific value for a certain variable. For instance, one may know how many people in the data set have a certain age and how many people in the data set are married, even though some values of the variable *Age* and the variable *Marital Status* are missing in an observed, but incomplete data set. In this article we assume that for several categories such frequencies are known, and our aim is to obtain a fully imputed data set that preserves these frequencies.

Note that if the known frequencies are available from administrative data, then our imputation methods will duplicate the distribution of administrative marginal totals in the completed data. Our imputation methods do not necessarily preserve the distributions in the reported data. In our evaluation study in Section 4 we will examine how well the distributions in the reported data are preserved.

In practice, it may happen that a variable is fully observed in the data set while at the same time a different total is known from another source. In that case either (at least) one of the sources contains errors, or the differences are caused by different concepts, different definitions, different moments of observation and so on. We recommend using statistical data editing and data integration techniques to correct these errors and other differences before proceeding with the imputation process (see De Waal et al. 2011, and ESSnet on Data Integration 2011, for an overview of statistical data editing and data integration techniques, respectively).

3. The Imputation Methods

3.1. The Basic Idea

The imputation methods we apply in this article are all based on a hot-deck donor approach. When hot-deck donor imputation is used, for each record containing missing values, the so-called recipient record, one uses the values of one or more other records, the so-called donor record(s), to impute these missing values.

Usually, hot-deck donor imputation is applied multivariately, that is several missing values in a record are imputed simultaneously, using the same donor record. For our problem this approach is less suited. If an imputed record fails the edits, all one can do is reject the donor record and use another donor record. For a relatively complicated set of edits, one may have to test many different potential donor records until a donor record is found that leads to an imputed record satisfying all edits. Moreover, for a relatively complicated set of edits one may not even be able to find a donor record for a certain recipient record such that the resulting imputed record satisfies all edits.

Even if we were able to find single donor records for all records requiring imputation, this would then solve only part of our problem, as the totals would only be preserved in very rare cases.

We therefore apply sequential univariate hot-deck donor imputation, where for each missing value in a record requiring imputation a different donor record may be selected. The variables with missing values are imputed sequentially. For each variable, the records for which the value of this variable is missing are imputed one by one. Once all records for this variable have been imputed, the next variable with missing values is considered. The univariate hot-deck imputation methods we apply are described in Subsection 3.2. These univariate hot-deck imputation methods are used to construct a list of possible donor values for a certain missing field. Whether a value is actually used to impute the missing field depends on whether the edits can be satisfied and the totals can be preserved.

While imputing a missing value, care is taken to ensure that the record can satisfy all edits. Only values of donor records that can result in a consistent record, that is a record that satisfies all edits, are eligible to be used. In Subsection 3.3 we explain how we determine whether a value is eligible to be used for imputation. For each record we make a list of values eligible for imputation for the variable under consideration.

An eligible value may only be used for actual imputation if the total can be preserved. Before an eligible value is actually used to impute a value, we first check whether the corresponding total can be preserved. If so, we use the value for imputation. If the total cannot be preserved, the value is rejected and the next value on the list of eligible values is selected. This process goes on until we find an eligible value such that the corresponding total can be preserved.

3.2. Univariate Hot-Deck Imputation Methods

In this article we apply two univariate hot-deck donor imputation methods: a nearest-neighbour approach and a random hot-deck approach.

3.2.1. Nearest-Neighbour Hot-Deck Imputation

Suppose we want to impute a certain variable v in a record r_0 using a pool of donors where the variable v is not missing. In the nearest-neighbour approach we calculate for each other record r in the pool of donors for which the value of v is not missing a distance given by

$$Dist\,(r_0, r) = \sum_{i \neq v} w_i\left(x_i^0, x_i^r\right), \tag{3}$$

where the sum is taken over all variables except variable v, x_i^0 denotes the value of the i-th variable in record r_0, x_i^r the value of the i-th variable in record r, and $0 \leq w_i\left(x_i^0, x_i^r\right) \leq 1$ a user-specified weight expressing how serious one considers a difference between x_i^0 and x_i^r to be. The weight $w_i\left(x_i^0, x_i^r\right)$ equals zero if $x_i^0 = x_i^r$. The weight $w_i\left(x_i^0, x_i^r\right)$ is large if one considers the difference between x_i^0 and x_i^r to be important, and small if one considers the difference to be unimportant. The value of the i-th variable in record r_0, x_i^0, or the value of the i-th variable in record r, x_i^r, may be missing. If x_i^0 or x_i^r is missing, we set $w_i\left(x_i^0, x_i^r\right)$ to 1.

To impute a missing value, we first select the potential donor value from the record with the smallest distance. If that value is allowed according to the edits (see Section 3.3), we put this value on an ordered list of potential donor values: the list of eligible values. If that value is not allowed according to the edits, we try the category corresponding to the record with the second smallest distance, and so on until we find a donor value that is allowed according to the edits. After all potential donor records have been checked for eligible values, we try all values not observed in the donor records (if any). Generally all possible values are observed in the donor records. However, in principle, some values may not be observed in the donor records and may be needed to satisfy the edits and preserve totals.

Note that, once a potential donor record has been checked, all subsequent records with the same value for v will give the same result for the check, and hence do not have to be checked.

As a remark, if we used the subset of variables that are observed for all records in (3) instead of the set of all variables, the potential donor records for a certain recipient record would be ordered in the same way for each variable with missing values. In that case, if possible, multivariate imputation, using several values from the first potential donor record on this list, would be used. Only if a value of the first potential donor record could not be used because this would lead to failed edits or nonpreserved totals, a value from another potential donor record would be used.

3.2.2. Random Hot-Deck Imputation

When random hot-deck imputation is applied, a random donor record is selected, often within certain subgroups defined by auxiliary data. In our case we use random hot-deck to construct a list of possible donor values for the missing field. Let K denote the number of categories of the variable to be imputed, and let R be the total number of records with an observed value for this variable. For each category c_k $(k = 1, \ldots, K)$ we determine the ratio p_k defined by the number of records for which the observed value for the variable to be imputed is equal to c_k divided by R. We then draw categories c_k $(k = 1, \ldots, K)$ without replacement with probabilities p_k $(k = 1, \ldots, K)$ in the donor population.

To impute a missing value, we first select the potential donor value that was drawn first. If that value is allowed according to the edits (see Subsection 3.3), we put this value on a list of potential donor values. If that value is not allowed, we try the potential donor value that was drawn second, and so on until we find a donor value that is allowed according to the edits. After all potential donor records have been checked for eligible values, we try all values not occurring in the donor records (if any) in a random order. Again, once a potential donor has been checked, all subsequent records with the same potential donor value will give the same result for the check and do not have to be checked anymore. As for nearest-neighbour imputation, we thus construct a list of potential donor values. For random hot-deck imputation, the exact order of the potential donor values is less important than for nearest-neighbour imputation. The important point here is that a list with all potential donor values is constructed.

3.3. Satisfying Edit Restrictions

In order to ensure that the set of edits can be satisfied, we derive so-called implied edits. These implied edits are necessary to guarantee that whenever we impute the current variable, the remaining variables can indeed be imputed in a manner consistent with the edits.

To determine the set of edits for the remaining variables to be imputed while imputing the current variable, we use the method proposed by Fellegi and Holt (1976) to eliminate a variable.

To eliminate a variable v_t, we start by determining all index sets S such that

$$\bigcup_{j \in S} F_t^j = Dom_t \tag{4}$$

and

$$\bigcap_{j \in S} F_i^j \neq \varnothing \quad \text{for } i \neq t. \tag{5}$$

From these index sets we select the *minimal* ones, that is the index sets S that obey (4) and (5), but none of whose proper subsets obey (4). Given such a minimal index set S we construct the implied edit

$$\bigcap_{j \in S} F_1^j \times \ldots \times \bigcap_{j \in S} F_{t-1}^j \times Dom_t \times \bigcap_{j \in S} F_{t+1}^j \times \ldots \times \bigcap_{j \in S} F_n^j.$$

By adding the implied edits resulting from all minimal sets S to the current set of edits and then removing all edits involving the eliminated variable, one obtains a set of edits for the remaining variables. It can be shown that if, and only if, this set of edits for the remaining variables can be satisfied, a value for the eliminated variable exists such that the original set of edits can be satisfied. We call this the lifting property, namely that the set of edits can be satisfied when a certain number of variables is "lifted" to a higher number of variables. The idea of the proof of the lifting property is that if a value does not exist for the eliminated variable such that the original set of edits can be satisfied, then one would be able to construct a violated implied edit, which would be a contradiction (see Fellegi and Holt 1976, and De Waal and Quere 2003, for details of the proof).

For records where multiple values are missing, we now order these variables in some order that we will describe in Subsection 3.5. Next, we eliminate the variables according to this order. Let us assume that, say, the values of variables v_1 to v_m are missing. We first substitute the values of the other variables into the original set of edits. This gives a set of edits E_0 that have to be satisfied by variables v_1 to v_m. We then eliminate variable v_1 from E_0 and obtain a set of edits E_1 that have to be satisfied by variables v_2 to v_m. Next, we eliminate variable v_2 from E_1 and obtain a set of edits E_2 that have to be satisfied by variables v_3 to v_m. We continue this process until we eliminate v_{m-1} from E_{m-2}, and obtain a set of edits E_{m-1} for variable v_m. For a single variable, edits simply define a set of allowed values for that variable. So, for variable v_m we now know which values are eligible for imputation. By a repeated application of the lifting property it can be shown that the original set of edits can be satisfied if and only if v_m satisfies E_{m-1}.

Once we have determined the edit sets $E_k\,(k = 0, \ldots, m-1)$, we can impute the variables in reverse order. That is, we try to impute v_m by means of one of our hot-deck imputation methods (see Subsection 3.2) until we have selected an eligible value that can also preserve the total for this variable (see Section 3.4). We fill in this value for v_m into the edits in E_{m-2}. This gives us a set of eligible values for variable v_{m-1}. We continue this procedure until we have imputed all variables. What is important here is that whenever we want to impute a certain variable in a certain record, we know the set of eligible values for that variable in this record. We will use this property to preserve totals (see Subsection 3.4).

Implied edits are often used to automatically identify erroneous fields in a data set (see Fellegi and Holt 1976). It is well known that the number of implied edits may be very large. In order to identify erroneous fields automatically, one basically has to generate implied edits for every possible subset of the variables. In our case, however, the number of implied edits is much less since we only have to consider a limited number of possible subsets as the variables are eliminated in a fixed order. For instance, if there are five variables, we would have to consider 32 subsets (ranging from eliminating no variables to eliminating all five variables) for identifying errors automatically in the Fellegi and Holt approach. For our method, we only need to examine six subsets (ranging from eliminating no variable, eliminating variable 1, eliminating variables 1 and 2, etc., to eliminating variables 1, 2, 3, 4 and 5).

Example: To illustrate the use of implied edits, we assume that we have a data set with the three variables *Marital Status*, *Age* and *Relation to Head of Household* and their categories defined in Subsection 2.1. We also assume that these variables have to satisfy edits (1) and (2). Now suppose that both *Marital Status* and *Age* in a certain record are missing, and that the value of *Relation to Head of Household* equals "Spouse". Suppose that we first impute *Age* and subsequently *Marital Status*. In this case we cannot simply ignore the edits involving the variable to be imputed later, *Marital Status*, while imputing *Age*, since it would be possible to impute the value "<16 years" for the missing value of *Age*, leading to no value for *Marital Status* such that all edits are satisfied.

The edits (1) and (2) imply the edit

$$(\{\text{Married, Unmarried, Divorced, Widowed}\}, \{< 16\,\text{years}\}, \{\text{Spouse}\}), \tag{6}$$

which expresses that someone who is less than 16 years of age cannot be the spouse of the head of household. This follows from (4) and (5) by taking $S = \{1, 2\}$ and eliminating variable *Marital Status* as explained here: The sets $F_i^j\,(i = 1, 2, 3; j = 1, 2)$ are given by $F_1^1 = \{\text{Married}\}$, $F_2^1 = \{< 16\,\text{years}\}$, $F_3^1 = \{\text{Spouse, Child, Other}\}$, $F_1^2 = \{\text{Unmarried, Divorced, Widowed}\}$, $F_2^2 = \{< 16\,\text{years}, \geq 16\,\text{years}\}$ and $F_3^2 = \{\text{Spouse}\}$. In order to eliminate variable *Marital Status*, we take the union of F_1^1 and F_1^2, and the intersections of $F_i^j\,(i = 2, 3; j = 1, 2)$.

If we take the implied edit in (6) into account while imputing the missing value for *Age*, we find that we cannot impute the value "< 16 years" and that only "≥ 16 years" is allowed. When "≥ 16 years" is imputed, *Marital Status* can indeed be imputed in a consistent manner.

Now that we have explained why implied edits are needed, we illustrate how we use them in our approach. Suppose we order the variables as follows: *Marital Status* and then *Age*. We substitute the value *of Relation to Head of Household* ("Spouse") into the edits (1) and (2), and obtain the edits

$$(\{\text{Married}\},\ \{< 16\,\text{years}\}) \tag{7}$$

and

$$(\{\text{Unmarried, Divorced, Widowed}\},\ \{< 16\,\text{years}, \geq 16\,\text{years}\}) \tag{8}$$

for *Marital Status* and *Age*. In this very simple case we now only have to eliminate one variable, *Marital Status*, and obtain the edit

$$(\{< 16\,\text{years}\}) \tag{9}$$

that has to be satisfied by *Age*. Edit (9) defines the set of eligible values for *Age:* in this case only the value "≥ 16 years" is allowed. If we impute "≥ 16 years" for the missing value of *Age*, we can be sure that a value for *Marital Status* exists such that all edits are satisfied. Imputing the value "≥ 16 years" for *Age* and substituting this value into edits (7) and (8) gives the edit

$$(\{\text{Unmarried, Divorced, Widowed}\})$$

for *Marital Status*. The set of allowed values for *Marital Status* hence consists of the value "Married" only.

3.4. Preserving Totals

In the previous subsection we have explained that whenever we want to impute a certain variable in a record we know the set of eligible values. For every record we now construct such a set of eligible values for the variable to be imputed. Suppose the variable to be imputed has K categories c_1 to c_k. We can then summarise the problem in a table as shown in Table 1 where N_{rec} is the number of records, a 0 denotes that the category is not eligible for imputation, a "$*$" that the category is eligible for imputation and a 1 that this value is observed (not missing) in the corresponding record. The $t_k\,(k = 1, \ldots, K)$ denote the known totals.

Table 1. Illustration of the sets of eligible values

	Cat. c_1	Cat. c_2	. . .	Cat. c_K
Record 1	*	0	. . .	*
Record 2	1	0	. . .	0
Record 3	0	*	. . .	*
. . .	. . .	. . .	. . .	. . .
Record N_{rec}	*	*	. . .	0
	t_1	t_2		t_k

Now, we impute the variable under consideration record by record. We select a value from the set of eligible values for the variable to be imputed for record 1. As explained in Subsection 3.2, the list of eligible values has been constructed using one of our hot-deck approaches. After a category c_x has been selected from the list of eligible values, we perform the following two checks:

1. Is the number of records that have been assigned to the selected category c_x less than the total t_x? If so, we perform the second check. If not, we reject the selected category c_x and select a new one.
2. Will it be possible to preserve the totals involving this variable if we accept the selected category c_x? If so, we accept this value, and go to the next record to be imputed. If not, we reject the selected category c_x and select a new one, which is again subjected to the same checks.

Checking whether the total can be preserved if we accept the selected category c_x is a well-known problem of combinatorial mathematics. It is called the "Harem problem" (see Anderson 1989). The "Harem problem" is a generalization of the "Marriage problem" (see e.g., Anderson 1989, and Van Lint and Wilson 2001). In the "Harem problem", several men (the categories in our case) can select a specified number (the t_k in our case) of wives (the records in our case) they are willing to marry (assign a record to a category in our case) and add to their "harem". For each category we make a list of records that can be assigned to this category (using the $*$'s and the 0's in Table 1). The 1's in Table 1 correspond to records in which categories have been observed, and hence have already been assigned to these categories.

A condition and a constructive algorithm for solving the "Harem problem" are given in Anderson (1989). The condition given by Anderson (1989) is: $t_k\,(k = 1, \ldots, K)$ records can be assigned to categories $c_k\,(k = 1, \ldots, K)$ if, and only if, for every subset $\{i_1, \ldots, i_m\}$ of $\{1, \ldots, K\}$ the lists of categories $c_{i_1}, \ldots, c_{i_m}$ contain in their union at least $t_{i_1} + \cdots + t_{i_m}$ records. This condition is hard to check directly. Fortunately, the constructive algorithm for solving the "Harem problem" described by Anderson (1989) provides a relatively simple way to check the condition and construct a solution at the same time. The underlying idea of this algorithm is to assign records to categories in a simple manner until one gets "stuck". Once that happens, a reshuffling algorithm (see the Appendix for a brief description of this algorithm, or Anderson 1989, for more details) is applied with the aim to assign one more record to the categories. This algorithm is repeatedly applied until either all records are assigned to categories, or until one again gets

"stuck". In the first case we have constructed a solution to this instance of the "Harem problem", and we have shown that it is possible to preserve the totals if we accept the selected category c_x. In the second case we have demonstrated that a solution to this instance of the "Harem problem" is not possible.

Note that if, for a certain variable to be imputed, the first record with a missing value has a solution to the "Harem problem", by construction all subsequent records to be imputed for that variable also have solutions to the "Harem problem".

Example: We illustrate the "Harem problem" and our approach to the imputation problem by means of a simple example. Suppose that for a certain variable to be imputed, we have summarised the problem in Table 2.

Now if we select category c_3 for the first record, the "Harem problem" for the remaining records turns out to be infeasible. This is easy to see: The remaining total of four records must be assigned to categories c_1 and c_2 in some way. However, record 3 cannot be assigned to either of these categories since a 0 denotes an ineligible category. This means that category c_3 must be rejected for record 1, and we have to impute category c_2 for this record. The "Harem problem" for the remaining records is then feasible. In fact, there is only one solution: Assign record 1 to category c_2, record 3 to category c_3, and records 2, 4 and 5 to category c_1.

3.5. Order of Imputing Variables and Records

In our evaluation study in Section 4 we have imputed the variables in increasing order of missing values. That is, we impute the variable(s) with the least number of missing values first, and end with the variable(s) with the most missing values. Possibly better orders for the variables to be imputed can be developed (see, e.g., Di Zio et al. 2004).

Obviously, for a given variable, for the first record it is generally easier to find solutions to the "Harem problem" than for later records. That is, for later records, one generally needs to try more potential donor values on average before one finds a value that satisfies edits and preserves the total (although one can be sure that such a value exists if the "Harem problem" has a solution for the first record). Since it may be difficult to find suitable imputation values for different variables of the same record, we randomize the records each time before we start imputing a new variable.

As noted in the previous subsection, for each new variable, it is only for the first record to be imputed that it may be impossible to find an imputation value that satisfies all edits and preserves the total. If we cannot find a suitable imputation value for that record, we

Table 2. An example of the "Harem problem"

	Cat. c_1	Cat. c_2	Cat. c_3
Record 1	0	*	*
Record 2	*	*	*
Record 3	0	0	*
Record 4	*	*	*
Record 5	*	0	*
	3	1	1

would have to backtrack. That is, we would have to return to a previously imputed variable, and impute one or more missing values for that variable in another way. This would lead to an extremely complicated and time consuming process.

By imputing the variable(s) with the least number of missing values first and the variable(s) with the most missing values last, we try to avoid having to backtrack. The later in the imputation process, the more difficult it is to satisfy all edits and preserve all totals. Therefore, by imputing the variables with the most missing values last, we try to make finding solutions for those variables a bit easier as the more values are missing, the more "freedom" one has to satisfy edits and to preserve the totals.

In addition, in order to avoid having to backtrack, we can also try to fill in values that deactivate edits at the start of the imputation process for variables to be imputed later, even if this leads to a slightly higher distance in (3) for the nearest-neighbour approach. For instance, edit (1) could be deactivated for *Relation to Head of Household* by filling in the value "Unmarried" for *Marital Status*. Instead of backtracking or deactivating edits one could also relax the problem by removing edits or by tolerating edits or totals to not be strictly satisfied. In our evaluation study described in Section 4, we did not have to backtrack or relax the problem. We did deactivate edits while imputing the first variable. For later variables we applied the usual approach described in Sections 3.1 to 3.4.

4. Evaluation Study

In this section we describe a study on a real data set to evaluate our imputation approaches. However, as the results may be influenced by the nonresponse mechanism, we ensure MCAR by artificially creating missingness.

4.1. Evaluation Data

The evaluation data set consists of observed data from the 2001 UK Census. The data set included 1,000 randomly selected households from one area. In the data set we have one record per person in the selected households. In total the data set contained 2,447 records. Each record contained six variables (the numbers of categories are in parenthesis): *Age* (4), *Ethnicity* (12), *Employment Status* (4), *Sex* (2), *Marital Status* (6) and *Relation to Head of Household* (10). In our evaluation study we assume that totals are known for all six variables.

For this data set three explicit categorical edits were defined:

- Someone whose age is less than 16 years cannot be employed.
- Someone whose age is less than 16 years cannot be married.
- Someone whose relation to the head of household is husband or wife has to be married.

The original data set for the 2001 UK Census did not contain any missing values. In this data set we randomly introduced fixed percentages of missing values using an MCAR mechanism where for each variable we created exactly the same percentages of missing values. We created ten replications of six data sets, each data set having a fixed percentage of missing values per variable: 1%, 2%, 5%, 10%, 20% and 90%. These data sets were imputed, using the imputation methods described in Section 3. The resulting imputed data

sets were subsequently compared to the original data. The evaluation measures used for this comparison are discussed in Subsection 4.3 and are calculated by averaging the evaluation measures calculated for each replication of the six data sets according to the percentage of missing values. As the nearest-neighbour imputation is deterministic in our implementation, all ten replications gave the same imputations. The evaluation measures for the random hot-deck method were relatively stable across the ten replicates.

Note that although we carried out ten replications of the imputation methods on each of the six data sets, our methods are in essence single imputation methods, rather than multiple imputation methods (see Rubin 1987). In practice, single imputation methods are preferred at NSIs rather than multiple imputation methods. In principle, our imputation methods can be adapted to multiple imputation to account for the extra variation arising from imputation.

4.2. The Imputation Methods

We evaluated two different imputation methods: one based on random hot-deck donor imputation and one based on nearest-neighbour hot-deck imputation. For the imputation method based on nearest-neighbour hot-deck imputation we have examined two versions. For both versions based on nearest-neighbour imputation, $w_i\left(x_i^0, x_i^r\right) = 0$ if $x_i^0 = x_i^r$ and $w_i\left(x_i^0, x_i^r\right) = 1$ if $x_i^0 \neq x_i^r$ for all variables except *Age* in the distance function (3). The two versions based on nearest-neighbour hot-deck imputation differ with respect to the weights used in the distance function (3) for variable *Age*.

In the distance function the values of *Age* are subdivided into four age groups. In one version of the method based on nearest-neighbour hot-deck imputation, $w_i\left(x_i^0, x_i^r\right) = 0$ if x_i^0 is in the same age group as x_i^r and $w_i\left(x_i^0, x_i^r\right) = 1$ if x_i^0 is in a different age group than x_i^r. This imputation method is referred to as the "equal nearest neighbour method". In the other version of the method based on nearest-neighbour hot-deck imputation, if, $w_i\left(x_i^0, x_i^r\right) = 0$ if x_i^0 is in the same age group as x_i^r, $w_i\left(x_i^0, x_i^r\right) = 0.25$ if x_i^0 and x_i^r differ by only one age group, $w_i\left(x_i^0, x_i^r\right) = 0.5$ if x_i^0 and x_i^r differ by two age groups, and $w_i\left(x_i^0, x_i^r\right) = 0.75$ if x_i^0 and x_i^r differ by three age groups. This imputation method is referred to as the "unequal nearest neighbour method".

4.3. Evaluation Results

The imputation methods are compared using the quality measures described as follows. Note that the measures are used as indicators where the smaller the value, the more the method is preferred.

Let T represent a frequency distribution for a two-way table produced from the data and let $T(r,c)$ be the frequency in the cell in row r and column c.[1] (In this section r and c refer to "row", respectively "column", instead of to "record" and "category" as in earlier sections.)

Distance metric: We use the Hellinger's Distance defined as:

$$HD\left(T_{orig}, T_{imp}\right) = \left\{0.5 \sum_{r,c} \left(\sqrt{T_{orig}(r,c)} - \sqrt{T_{imp}(r,c)}\right)^2\right\}^{1/2}$$

with *orig* and *imp* referring to the original and imputed tables respectively. The *HD*

provides a measure of similarity between two probability distributions typically used for positive or zero counts.

Impact on measure of association: The first measure of association is defined as the per cent difference in the Cramer's V statistic as:

$$RCV\left(T_{orig}, T_{imp}\right) = \frac{100 \times \{CV\left(T_{imp}\right) - CV\left(T_{orig}\right)\}}{CV\left(T_{orig}\right)}$$

where

$$CV\left(T\right) = \sqrt{\frac{\chi^2}{\min\left(N_R - 1, N_C - 1\right)}}$$

is the Cramer's V measure of association defined in terms of χ^2, the usual Pearson chi-squared statistic for testing independence in a two-way table, N_R is the number of rows and N_C is the number of columns. The RCV provides a measure of attenuation of the association in the table.

The second measure of association is defined as the per cent difference in the variance of the cell counts:

$$RV\left(T_{orig}, T_{imp}\right) = \frac{100 \times \{V\left(T_{imp}\right) - V\left(T_{orig}\right)\}}{V\left(T_{orig}\right)}$$

where

$$V\left(T\right) = \frac{\sum_{r,c}(T(r,c) - \bar{T})^2}{N_R N_C - 1}.$$

The RV provides a measure of attenuation of the counts in the table indicating whether the cell counts are "flattening" as a result of the imputation.

Impact on an ANOVA analysis: Another form of bivariate analysis consists of comparing proportions in a category of a column (outcome) variable between categories of a row (explanatory) variable. Let

$$P^c(r) = \frac{T(r,c)}{\sum_c T(r,c)}$$

be the proportion in column c for row r and define the between-row variance of this proportion by:

$$BV(P^c) = \frac{\sum_r (P^c(r) - P^c)^2}{N_R - 1} \text{ where } P^c = \frac{\sum_r T(r,c)}{\sum_{r,c} T(r,c)}.$$

The measure is defined as:

$$BVR\left(P^c_{orig}, P^c_{imp}\right) = \frac{100 \times \left\{BV\left(P^c_{imp}\right) - BV\left(P^c_{orig}\right)\right\}}{BV\left(P^c_{orig}\right)}$$

The BVR provides a measure of attenuation of between group differences in an ANOVA analysis and indicates the undesirable result that the group proportions are "flattening" towards the overall proportion.

Figures 1 through 4 present graphs of the average quality measures across the ten replicates for some main distributions in the data set. The unequal nearest neighbour method provided similar results to the equal nearest neighbour method and hence we compare the random hot-deck method (denoted by "random") with the equal nearest neighbour method (denoted by "equal_nn") in the figures.

Figure 1a presents the Hellinger's Distance (*HD*) on a table of counts spanned by *Age Group* and *Employment Status* (16 cells). For all imputation rates, the equal nearest

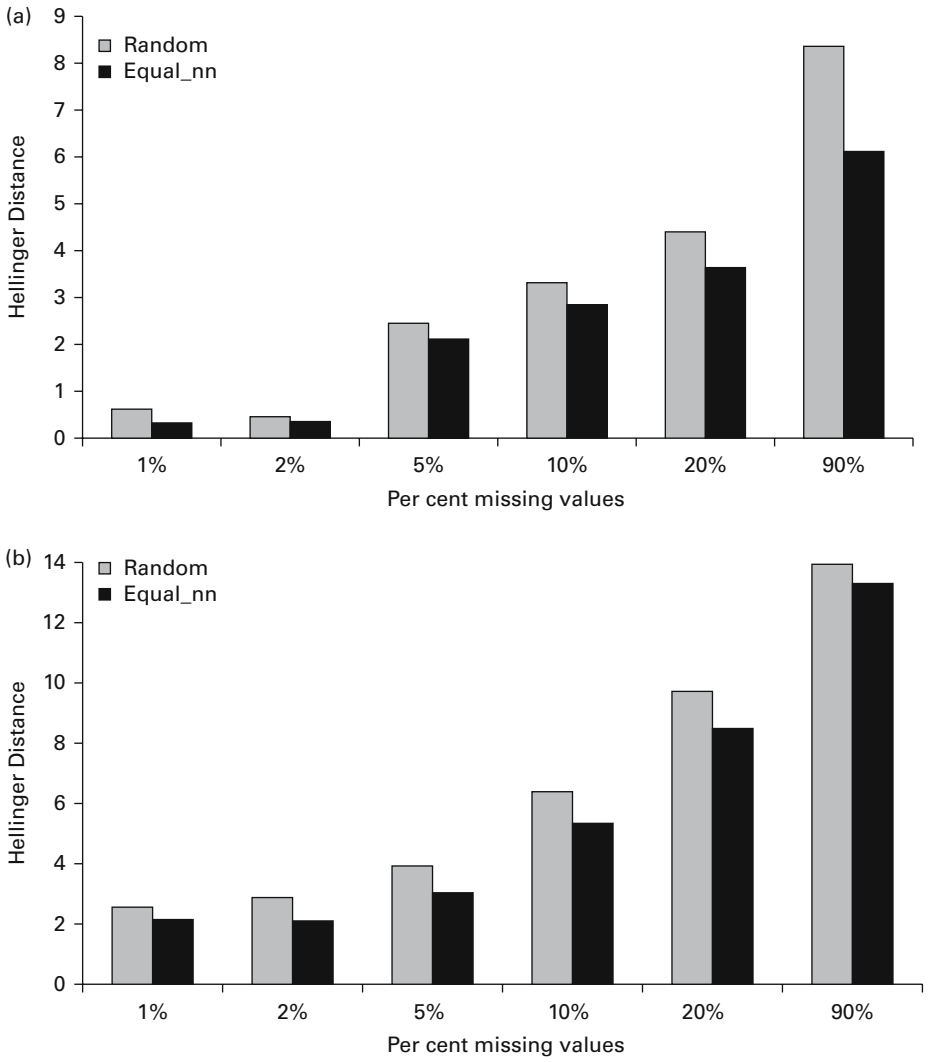

Fig. 1. (a) Average Hellinger's Distance (HD) across replicates on the table Age Group and Employment Status. (b) Average Hellinger's Distance (HD) across replicates on the table Age Group and Relation to Head of Household

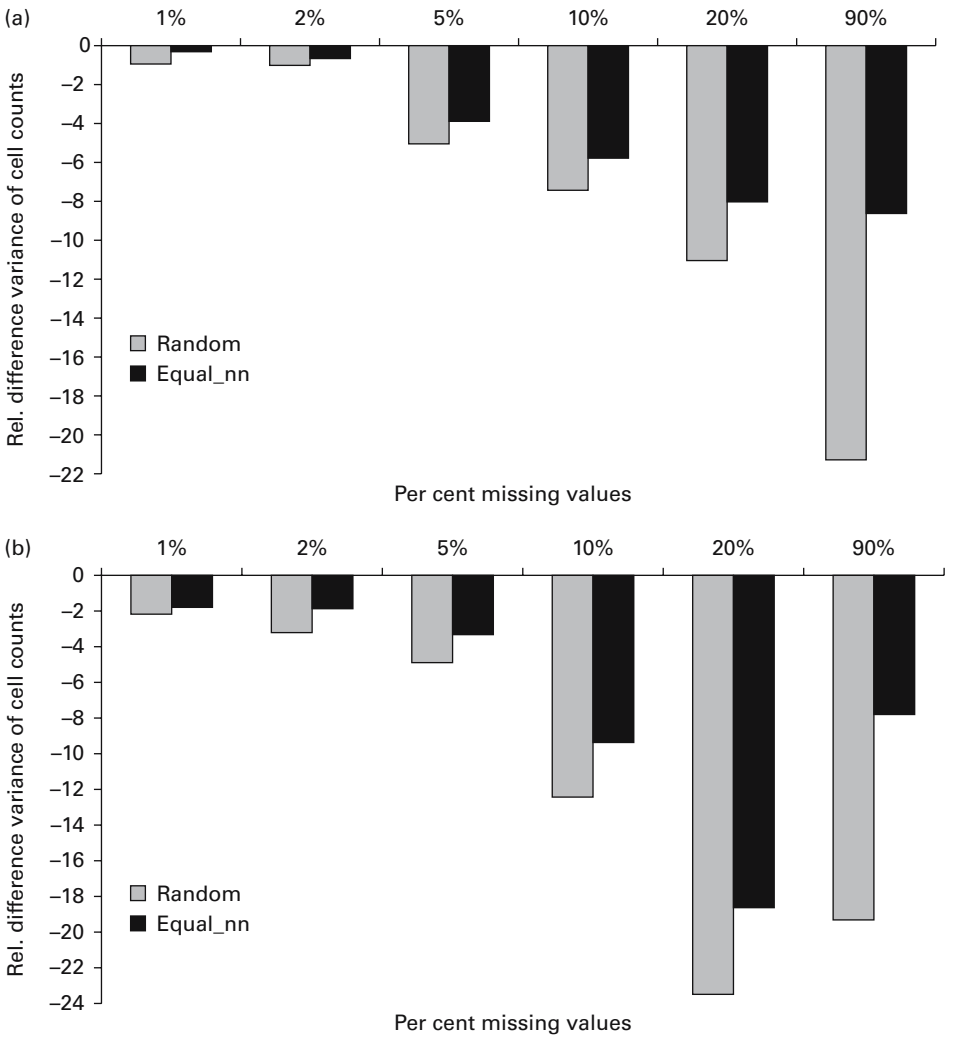

Fig. 2. (a) Average per cent relative difference in variance of cell counts (RV) across replicates on the table Age Group and Employment Status. (b) Average per cent relative difference in variance of cell counts (RV) across replicates on the table Age Group and Relation to Head of Household

neighbour method has lower Hellinger's Distance compared to the random method. Figure 1b presents the Hellinger's Distance for the table spanned by *Age Group* and *Relation to Head of Household* (40 cells) showing similar results.

Figure 2a presents the per cent relative difference in the variance of the cell counts for the table spanned by *Age Group* and *Employment Status*. The negative values of the *RV* measure means that the variance of counts with imputed values is less than the original variance of counts. The cell counts are "flattened" as a result of the imputation, leading to a smaller variance of the counts. The equal nearest neighbour method (as well as the unequal nearest neighbour method) has less change in the variance of the cell counts compared to the random method. Figure 2b presents the *RV* measure for the table spanned by *Age Group* and the *Relation to the Head of Household* with similar results.

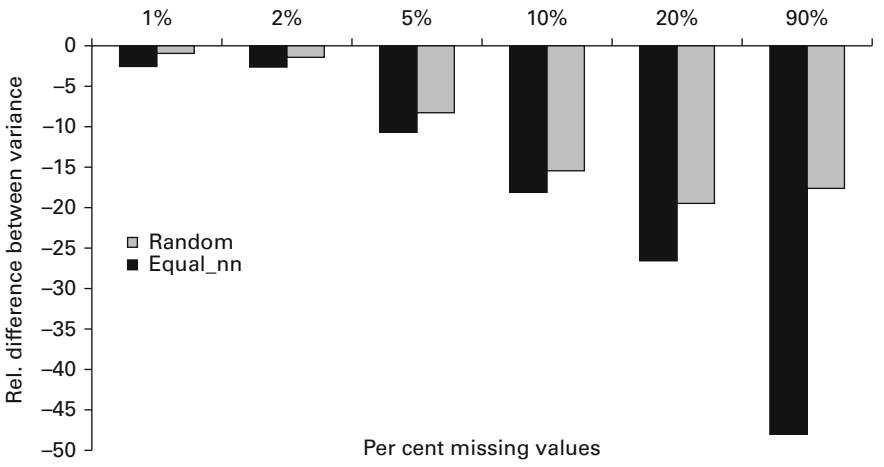

Fig. 3. Average per cent relative difference in between variance (BVR) across replicates of proportion of Employed across Sex and Age Groups

Figure 3 presents the per cent relative difference in the between variance of the proportion of employed persons in groups defined by *Sex* and *Age* groups (*BVR*). The negative values of the *BVR* measure means that the between variance of the group proportions of employed persons with imputed values is less than the original between variance. The group proportions are attenuating to the overall proportion as a result of the imputation. Again, equal nearest neighbour method (and the unequal nearest neighbour method) has less change in the *BVR* compared to the random method.

Figure 4a presents the per cent relative difference in the Cramer's V statistic of the table spanned by *Age Groups* and *Employment Status* (*RCV*). The negative values of the *RCV* measure means that the Cramer's V statistic on the table with imputed values is less than the original Cramer's V statistic. The table of counts is attenuating towards assumptions of independence compared to the original table. For all imputation rates, the equal nearest neighbour method has less change in the Cramer's V statistic than the random method and similarly for the unequal nearest neighbour method. Figure 4b presents the *RCV* measure for the table spanned by *Age Groups* and *Relation to Head of Household* with similar results.

In Figure 5, we present box plots of the proportion of values that were *not* imputed back to their original value in the data set according to the percentage missing and imputation method. Each box plot includes a total of 38 proportions which is the number of categories of the six variables in the data set. The proportions were calculated as the average across the replications. The proportion is very small for the data sets, with 1% and 2% missing values. Based on the data sets with 5% missing values and over, we can see a slight advantage to the equal nearest neighbour approach with less outlying proportions, a smaller interquartile range of the proportions and a slightly smaller median proportion.

5. Discussion

In this article we have developed two imputation methods for categorical data that take edits and known totals into account while imputing a record. One of the imputation

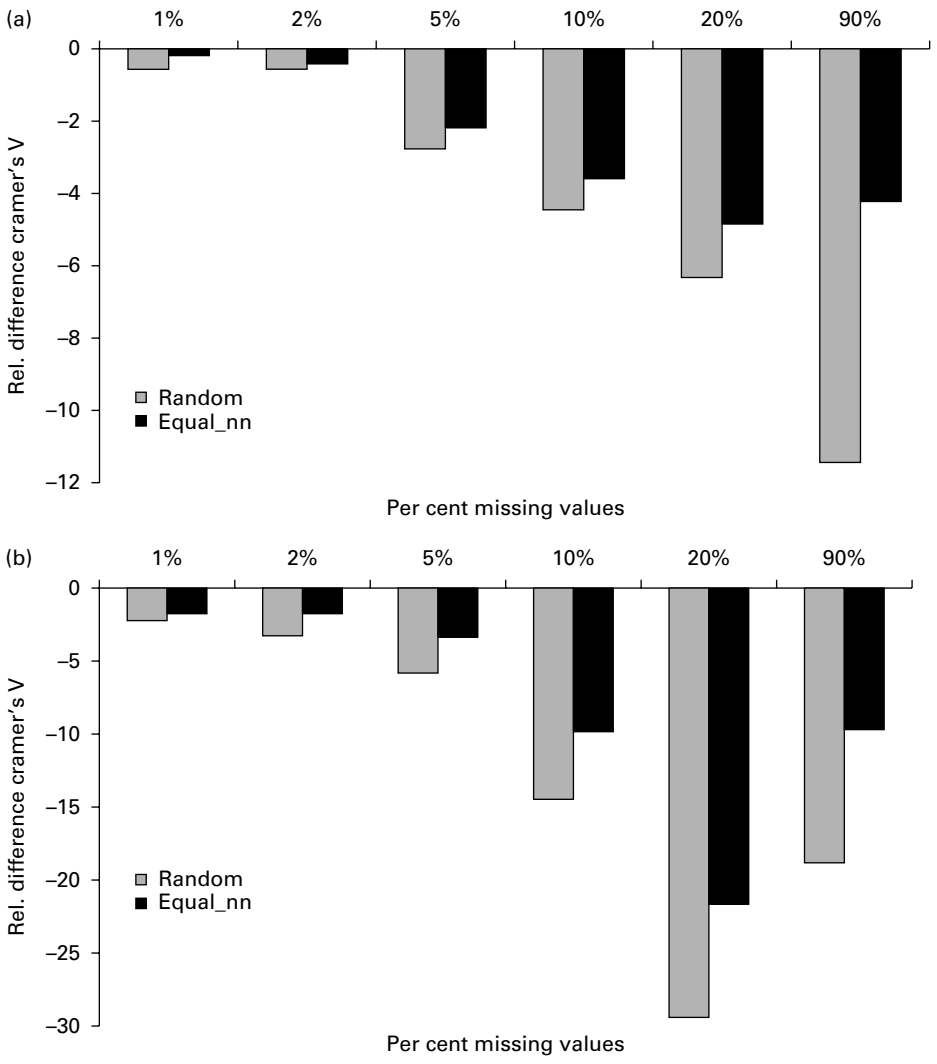

Fig. 4. (a) Average per cent relative difference in Cramer's V across replicates on the table of Age Groups and Employment Status. (b) Average percent relative difference in Cramer's V across replicates on the table of Age Groups and Relation to the Head of Household

methods proposed in this article is based on random hot-deck donor imputation and the other on nearest-neighbour donor imputation. Our evaluation study shows that the method based on nearest-neighbour imputation performs slightly better than the method based on random imputation. In our evaluation study, changing the weights in the distance function of the method based on nearest neighbour imputation had little or no effect on the outcome of the results. All imputation methods provide exactly the totals to those used in the benchmarking. For non-benchmarked subdomain totals, one can assess the potential bias as shown by the Hellinger's Distance in Figures 1a and 1b. To ensure totals for subdomains of interest, the imputation methods can be carried out separately in each subdomain assuming that the totals are known.

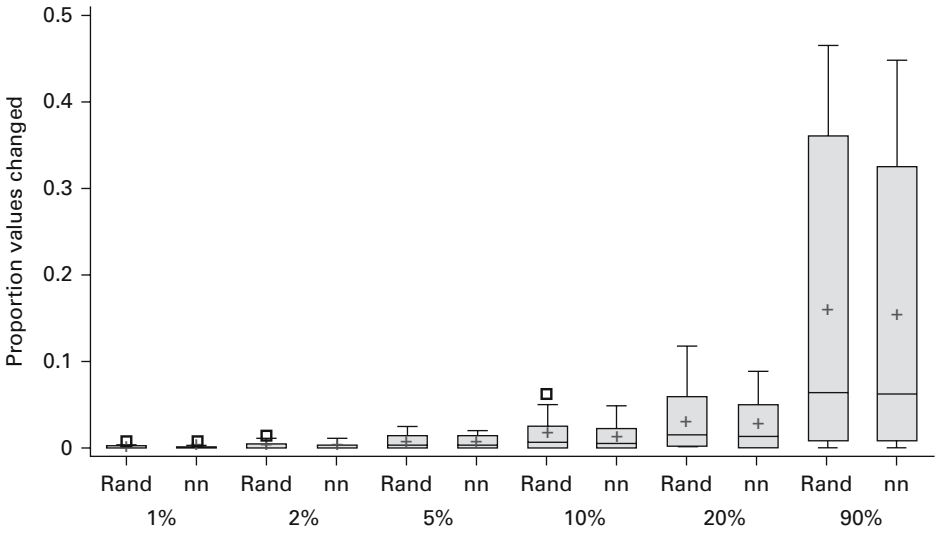

Fig. 5. Proportion of values changed in the data set for the Random (rand) and Equal Nearest Neighbour (nn) approaches according to per cent missing values (average across replicates)

The problem of imputing missing data while satisfying edits and preserving totals has hardly been studied in the literature. Our methods are among the first for this kind of problem. Many aspects of the developed methods can undoubtedly be extended and improved upon.

A possible extension is to develop similar methods for the situation where one wants to impute a sample data set, instead of all units in the population as in the current article. In order to impute a sample data set so that population totals are preserved, one would have to extend our methods to deal with sampling weights. If all sampling weights are integers, a first idea would be to simply make w copies of a record with sampling weight w, and then apply the methods described in this article. When translating this back to the sample, fractions of categories would then be "imputed" in each record. If sampling weights are not integers, the situation is more complicated, and one would have to do some rounding. It is very likely that more efficient and better approaches can be developed for extending our methods to sample data sets.

Another interesting extension is to develop similar imputation methods for the case where bivariate marginal distributions with overlapping variables, say of the pair of variables (X,Y) and the pair of variables (X,Z), are known instead of only univariate marginal distributions. In principle, this could be solved by constructing the crossings of (X,Y) and of (X,Z), and adding these crossings to the set of variables. In order to avoid any inconsistencies between the marginals of these crossings and the marginals of variables X, Y and Z, one would then need to add edits, for example: "if $(X = x, Y = y)$ then $(X = x)$" and "if $(X = x, Y = y)$ then $(Y = y)$" for the crossing of X and Y, and similar edits for the crossing of Y and Z.

Although this is, in principle, a possible approach, it is likely to be time consuming with more chances of getting "stuck" in the "Harem problem" and having to backtrack. A more efficient approach for this situation remains to be developed.

Alternatively, knowing the marginal of (*X*,*Y*) and (*X*,*Z*), one could estimate (*X*,*Y*,*Z*) using log-linear modelling and carry out the imputation separately in each subdomain of this cross-classification. Again, it is likely that better approaches can be developed.

It is unclear whether the use of known totals in the imputation process preserves correlations better between variables compared to when totals are not used in the imputation process. We hope to explore this in future research.

Our imputation methods consist of two different parts: a statistical part (drawing potential donor values) and a combinatorial part (satisfying edits and preserving totals). The final aim of research in this area should be to develop a statistical framework that organically incorporates the combinatorial part as well.

Appendix: The Reshuffling Algorithm for the "Harem Problem"

Assume that (some) records have already been assigned to categories by means of a simple algorithm, for example by a "greedy" algorithm where first as many records as possible are assigned to the first category without exceeding the total for this category, then as many records as possible out of the remaining records are assigned to the second category without exceeding the total for that category, and so on, until either all records have been assigned to categories or one gets stuck. In the first case, the "Harem problem" has been solved. In the second case, we apply the reshuffling algorithm sketched below, which aims to assign one extra record to the categories.

As in Subsection 3.4, we denote the number of records by N_{rec}. Define $L(r_i)$ as the set of categories that are eligible for imputation of record r_i ($i = 1, \ldots, N_{rec}$). With $r_{[j]}$ we denote the j-th record that is selected in the procedure sketched below. For example, if the first record selected is r_3, then $r_{[1]} = r_3$ and $L(r_{[1]}) = L(r_3)$. The same record may be selected several times, so some of the $r_{[j]}$ may refer to the same record. Likewise, we use $c_{[j]}$ to denote the j-th category that is selected in the procedure, for example if the first category selected is c_3 then $c_{[1]} = c_3$. Again, the same category may be selected several times, so some of the $c_{[j]}$ may refer to the same category.

1. Select a record $r_{[1]}$ that has not yet been assigned to a category.
2. Select a category $c_{[1]}$ from $L(r_{[1]})$.
   - If $r_{[1]}$ may be assigned to $c_{[1]}$ without exceeding the total for this category, we are obviously done.
   - If $r_{[1]}$ may not be assigned to $c_{[1]}$, we set $L(r_{[1]}) := L(r_{[1]}) - \{c_{[1]}\}$, i.e., $c_{[1]}$ is dropped from $L(r_{[1]})$. Go to Step 3.
3. Select a record $r_{[2]}$ that has been assigned to $c_{[1]}$, and set $L(r_{[2]}) := L(r_{[2]}) - \{c_{[1]}\}$.
4. Select a category $c_{[2]}$ from $L(r_{[2]})$.
   - If $r_{[2]}$ may be assigned to $c_{[2]}$ without exceeding the total for this category, we are done (see below).
   - If $r_{[2]}$ may not be assigned to $c_{[2]}$, we set $L(r_{[2]}) := L(r_{[2]}) - \{c_{[2]}\}$ and go to Step 5.
5. Select a record $r_{[3]}$ that has been assigned to $c_{[2]}$, and set $L(r_{[3]}) := L(r_{[3]}) - \{c_{[2]}\}$.
6. And so on.

This reshuffling algorithm will eventually terminate. It can terminate in two possible ways:

Table A.1. Preliminary assignment of records to categories

	Cat. c_1	Cat. c_2	Cat. c_3
Record 1	0	$\underline{1}$	$\underline{0}$
Record 2	$\underline{0}$	$\underline{0}$	$\underline{1}$
Record 3	0	0	$\underline{0}$
Record 4	$\underline{1}$	$\underline{0}$	$\underline{0}$
Record 5	$\underline{1}$	0	$\underline{0}$
	3	1	1

a. We can assign some $r_{[k]}$ to a category $c_{[k]}$. In this case we can assign an extra record to a category. Namely, we can assign $r_{[k]}$ to $c_{[k]}$. Previously, $r_{[k]}$ had been assigned to a category $c_{[m]}$ ($m \leq k-1$). To this $c_{[m]}$ we can assign a record $r_{[p]}$ ($p \leq m$). We can continue in this way until we can assign record $r_{[1]}$ to category $c_{[1]}$.
   At this moment we have assigned an extra record to a category, and we are ready to restart the algorithm with another record that has not yet been assigned to a category. When there are no more records that need to be assigned to a category, this instance of the "Harem problem" has been solved.
b. We try to select a category from an empty set $L(r_{[j]})$. In this case we can conclude that this instance of the "Harem problem" is unsolvable.

We illustrate the above algorithm on the "Harem problem" given in Table 2. We assume that some records have already been assigned to categories by means of a simple "greedy" algorithm. The preliminary assignment of records to categories after application of this "greedy" algorithm is summarised in Table A.1, where categories that are eligible for imputation are underlined.

$L(r_1) = \{c_2, c_3\}$, $L(r_2) = \{c_1, c_2, c_3\}$, $L(r_3) = \{c_3\}$, $L(r_4) = \{c_1, c_2, c_3\}$ and $L(r_5) = \{c_1, c_3\}$. Only r_3 has not yet been assigned to a category, so we select $r_{[1]} = r_3$. We select $c_{[1]} = c_3$ from $L(r_3)$, and update $L(r_3) := \emptyset$. We select a record $r_{[2]}$ that has been assigned to c_3. In this case there is only one option, namely record r_2, so, $r_{[2]} = r_2$ and we update $L(r_2) := \{c_1, c_2\}$. We select a category, say $c_{[2]} = c_2$, from $L(r_2)$, and update $L(r_2) := \{c_1\}$. We select a record $r_{[3]}$ that has been assigned to c_2. In this case there is again only one option, namely record r_1, so, $r_{[3]} = r_1$, and we update $L(r_1) := \{c_3\}$. We select $c_{[3]} = c_3$ from $L(r_1)$, and update $L(r_1) := \emptyset$. We select a record $r_{[4]}$ that has been assigned

Table A.2. Assignment of records to categories after the reshuffling algorithm

	Cat. c_1	Cat. c_2	Cat. c_3
Record 1	0	$\underline{1}$	$\underline{0}$
Record 2	$\underline{1}$	$\underline{0}$	$\underline{0}$
Record 3	0	0	$\underline{1}$
Record 4	$\underline{1}$	$\underline{0}$	$\underline{0}$
Record 5	$\underline{1}$	0	$\underline{0}$
	3	1	1

to c_3. In this case there is again only one option, namely record r_2, so, $r_{[4]} = r_2$. Updating $L(r_2)$ has no effect: $L(r_2) := \{c_1\}$. We select $c_{[4]} = c_1$, from $L(r_2)$.

Record $r_{[4]} = r_2$ can be assigned to $c_{[4]} = c_1$. Previously, r_2 had been assigned to category $c_{[1]} = c_3$. In turn, we can assign record $r_{[1]} = r_3$ to category $c_{[1]} = c_3$. The assignment of records to categories after the reshuffling algorithm is summarised in Table A.2.

In this case, the "Harem problem" has been solved. In general one needs to apply the reshuffling algorithm several times before the "Harem problem" is solved, or before one can conclude that this instance of the problem is unsolvable.

6. References

Anderson, I. (1989). A First Course in Combinatorial Mathematics, (second edition). Oxford: Oxford University Press.

De Waal, T., Pannekoek, J., and Scholtus, S. (2011). Handbook of Statistical Data Editing and Imputation. New York: John Wiley & Sons.

De Waal, T. and Quere, R. (2003). A Fast and Simple Algorithm for Automatic Editing of Mixed Data. Journal of Official Statistics, 19, 383–402.

Di Zio, M., Scanu, M., Coppola, L., Luzi, O., and Ponti, A. (2004). Bayesian Networks for Imputation. Journal of the Royal Statistical Society: Series A, 167, 309–322.

ESSnet on Data Integration (2011). Report on WP 2: Methodological Developments. Available at: http://www.essnet-portal.eu/sites/default/files/131/WP2.pdf.

Favre, A.-C., Matei, A., and Tillé, Y. (2005). Calibrated Random Imputation for Qualitative Data. Journal of Statistical Planning and Inference, 128, 411–425.

Fellegi, I.P. and Holt, D. (1976). A Systematic Approach to Automatic Edit and Imputation. Journal of the American Statistical Association, 71, 17–35.

Houbiers, M. (2004). Towards a Social Statistical Database and Unified Estimates at Statistics Netherlands. Journal of Official Statistics, 20, 55–75.

Kalton, G. and Kasprzyk, D. (1986). The Treatment of Missing Survey Data. Survey Methodology, 12, 1–16.

Knottnerus, P. and Van Duin, C. (2006). Variances in Repeated Weighting with an Application to the Dutch Labour Force Survey. Journal of Official Statistics, 22, 565–584.

Little, R.J.A. and Rubin, D.B. (2002). Statistical Analysis with Missing Data (second edition). New York: John Wiley & Sons.

Liu, T.-P. and Rancourt, E. (1999). Categorical Constraints Guided Imputation for Nonresponse in Survey. Report, Statistics Canada.

Longford, N.T. (2005). Missing Data and Small-Area Estimation. New York: Springer.

McKnight, P.E., McKnight, K.M., Sidani, S., and Figueredo, A.J. (2007). Missing Data – A Gentle Introduction. New York: The Guilford Press.

Pannekoek, J., Shlomo, N., and De Waal, T. (2008). Calibrated Imputation of Numerical Data under Linear Edit Restriction. UN/ECE Work Session on Statistical Data Editing, Vienna.

Pfefferman, D. and Rao, C.R. (2009). Handbook of Statistics 29, Volume 29A. Amsterdam: Elsevier.

Rubin, D.B. (1976). Inference and Missing Data. Biometrika, 63, 581–592.
Rubin, D.B. (1987). Multiple Imputation for Nonresponse in Surveys. New York: John Wiley & Sons.
Schafer, J.L. (1997). Analysis of Incomplete Multivariate Data. London: Chapman & Hall.
Van Lint, J.H. and Wilson, R.M. (2001). A Course in Combinatorics (second edition). Cambridge: Cambridge University Press.
Winkler, W.E. (2003). Contingency-Table Model for Imputing Data Satisfying Analytic Constraints. U.S. Bureau of the Census, Washington, D.C.

Received April 2012
Revised February 2013
Accepted February 2013