Hot deck imputation of numerical data under edit restrictions

Wieger Coutinho and Ton de Waal

The views expressed in this paper are those of the author(s) and do not necessarily reflect the policies of Statistics Netherlands

Discussion paper (201223)

Statistics Netherlands

The Hague/Heerlen, 2012

Explanation of symbols

.	data not available
*****	provisional figure
******	revised provisional figure (but not definite)
x	publication prohibited (confidential figure)
–	nil
–	(between two figures) inclusive
0 (0.0)	less than half of unit concerned
empty cell	not applicable
2011–2012	2011 to 2012 inclusive
2011/2012	average for 2011 up to and including 2012
2011/’12	crop year, financial year, school year etc. beginning in 2011 and ending in 2012
2009/’10–2011/’12	crop year, financial year, etc. 2009/’10 to 2011/’12 inclusive

Due to rounding, some totals may not correspond with the sum of the separate figures.

Publisher
Statistics Netherlands
Henri Faasdreef 312
2492 JP The Hague

Prepress
Statistics Netherlands
Grafimedia

Cover
Teldesign, Rotterdam

Information
Telephone +31 88 570 70 70
Telefax +31 70 337 59 94
Via contact form:
www.cbs.nl/information

Where to order
E-mail: verkoop@cbs.nl
Telefax +31 45 570 62 68

Internet
www.cbs.nl

ISSN: 1572-0314

Hot deck imputation of numerical data under edit restrictions

Wieger Coutinho and Ton de Waal

Summary: A common problem faced by statistical institutes is that data may be missing from collected data sets. The typical way to overcome this problem is to impute the missing data. The problem of imputing missing data is complicated by the fact that statistical data collected by statistical institutes often have to satisfy certain edit rules, which for numerical data usually take the form of linear restrictions. Standard imputation methods for numerical data as described in the literature generally do not take such linear edit restrictions on the data into account. Hot-deck imputation techniques form a well-known class and relatively simple to apply class of imputation methods. In this paper we extend this class of imputation methods so that linear edit restrictions are satisfied.

Keywords: Fourier-Motzkin elimination, hot deck imputation, linear edit restrictions

1. Introduction

A well-known problem occurring in surveys is that data may be missing. Some population units that were asked to reply to a questionnaire may not have responded at all. This is referred to as unit nonresponse. In many other cases a respondent did answer some questions, but not all. This is referred to as item nonresponse. There may be various reasons for not answering a certain question: the respondent may not understand the question, may not know the answer to the question, may forget to answer the question, may refuse to answer the question because he considers the answer to the question as too confidential, and so on. In this paper we focus on item nonresponse for numerical, more precisely: continuous, data. Whenever we refer to missing data in this paper, we will mean missing data due to item nonresponse.

The most common solution to handle missing data in data sets is imputation, where missing values are estimated and filled in. The main problem of imputation is to recover the statistical aspects of the true data, such as means, totals, and (co)variances, as well as possible. This problem has been amply studied and described in the literature. See, for instance, Kalton and Kasprzyk (1986), Rubin (1987), Kovar and Whitridge (1995), Schafer (1997), Little and Rubin (2002), Longford (2005), and De Waal, Pannekoek and Scholtus (2011).

At National Statistical Institutes (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 the profit and the costs of an enterprise have to sum up to its turnover, and that the turnover of an enterprise should be at least zero. Records that do not satisfy these edits are inconsistent, and are hence considered incorrect. While imputing a record, we aim to take these edits into account, and thus ensure that the final, imputed data satisfy all edits.

Ensuring that the imputed data satisfy edits is especially important for NSIs as NSIs have the responsibility to supply data for many different, both academic and non-academic, users in society. For the majority of these users, inconsistent data are incomprehensible. They may reject the data as being an invalid source or make adjustments themselves. This hampers the unifying role of NSIs in providing data that are undisputed by different parties such as policy makers in government, opposition, trade unions, employer organizations, etc. (See also Pannekoek and De Waal, 2005.) As stated by Särndal and Lundström (2005, p. 176): "Whatever the imputation method used, the completed data should be subjected to the usual checks for internal consistency. All imputed values should undergo the editing checks normally carried out for the survey".

Although much research on imputation techniques has been carried out, imputation under edits is still a rather neglected area of research. As far as we are aware, apart from some research at NSIs (see, e.g., Tempelman, 2007; Coutinho, De Waal and Remmerswaal, 2011; De Waal, Pannekoek and Scholtus, 2011) hardly any research on general approaches to imputation under edit restrictions has been carried out. An exception is imputation based on a truncated multivariate normal model (see, e.g., Geweke, 1991, and Tempelman, 2007). Some software packages developed by NSIs, such as GEIS (Kovar and Whitridge, 1990), SPEER (Winkler and Draper, 1997) and SLICE (De Waal, 2001), also ensure that edits are satisfied after imputation. The techniques for ensuring that the edits are satisfied implemented in these software packages are generally ad-hoc methods, such as using pro-rating and afterwards adjusting imputed values so that these adjusted values satisfy the edits.

Simple sequential imputation of the missing data, where edits involving fields that have to be imputed subsequently are not taken into account while imputing a field, may lead to inconsistencies. Consider, for example, a record where the values of two variables, x and y, are missing. Assume these variables have to satisfy three edits saying that x is at least 50, y is at most 100, and y is greater than or equal to x. Now, if x is imputed first without taking the edits involving y into account, one might impute the value 150 for x. The resulting set of edits for y, i.e. y is at most 100 and y is greater than or equal to 150, cannot be satisfied. Conversely, if y is imputed first without taking the edits involving x into account, one might impute the value 40 for y. The resulting set of edits for x, i.e. x is at least 50 and 40 is greater than or equal to x, cannot be satisfied.

In this paper we develop an extension of the class of hot deck imputation techniques (see Andridge and Little, 2010) so that edits are satisfied. Hot deck imputation, where missing values are imputed with values observed in other records in the same data set, is a well-known and at the same time relatively simple way to impute missing data. The simplicity of hot deck imputation makes it an attractive technique to implement and apply in practice. The main aim of our paper is to illustrate how the class of hot deck imputation techniques can be extended so that edits are satisfied, rather than developing “optimal” imputation methods for the data sets examined in our evaluation study.

The remainder of this paper is organised as follows. Section 2 first discusses the kind of linear edits on which we will focus in this paper. Section 3 discusses our extension of hot deck imputation. An evaluation study is described in Section 4. Section 5 concludes the paper with a short discussion.

2. Linear edit restrictions

In this paper we focus on linear edits for numerical data. Linear edits are either linear equations or linear inequalities. We denote the number of continuous variables by *n*, and the variables by x_i (*i*=1,…,*n*). We assume that edit *j* (*j*=1,...,*J*) can be written in either of the two following forms:

$$a_{1j}x_1 + \ldots + a_{nj}x_n + b_j = 0\,, \tag{2.1a}$$

or

$$a_{1j}x_1 + \ldots + a_{nj}x_n + b_j \geq 0\,. \tag{2.1b}$$

Here the a_{ij} and the b_j are certain constants, which define the edit.

Edits of type (2.1a) are referred to as balance edits. An example of such an edit is

$$T - C = P, \tag{2.2}$$

where *T* is the turnover of an enterprise, *P* its profit, and *C* its costs. Edit (2.2) expresses that the profit and the costs of an enterprise should sum up to its turnover. A record not satisfying this edit is obviously incorrect. Edit (2.2) can be written in the form (2.1a) as $T - P - C = 0$.

Edits of type (2.1b) are referred to as inequality edits. An example is

$$T \geq 0\,, \tag{2.3}$$

expressing that the turnover of an enterprise should be non-negative. An inequality edit such as (2.3), expressing that the value of a variable should be non-negative, is also referred to as a non-negativity edit.

3. Extensions of hot deck imputation

3.1 The basic idea

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

Usually, hot deck imputation is applied multivariately, i.e. several missing values in a record are imputed simultaneously, using the same donor record. For our problem that approach is not always feasible. If an imputed record fails the edits, all one could do in such an approach would be to 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 some recipient records such that the resulting imputed records satisfy all edits.

Instead of applying multivariate hot deck imputation, we therefore apply sequential univariate hot deck imputation, where for each missing value in a record requiring imputation in principle a different donor record may be selected. The variables with missing values are imputed sequentially. If possible, a single donor record is used for all missing values in a recipient, though. The univariate hot deck imputation methods we apply are described in Subsection 4.3. These univariate hot deck imputation methods are used to order the possible 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. In our application, the variables are imputed in increasing order of number of missing values, i.e. the variable with the least missing values is imputed first, the variable with the most missing values last.

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, i.e. a record that satisfies all edits, are used for imputation. How we ensure that edits can be satisfied is explained in Section 3.2.

3.2 Satisfying edits

As illustrated in the Introduction, simple sequential imputation of the missing data may result in violated edits. To ensure that sequential imputation leads to satisfied edits, we eliminate the variables to be imputed by means of Fourier-Motzkin elimination. Fourier-Motzkin elimination (see Duffin, 1974; De Waal, Pannekoek and Scholtus, 2011; and the Appendix of the current paper) is a technique to project a set of linear constraints involving *m* variables onto a set of linear constraints involving *m*-1 variables. The original set of constraints involving *m* variables can be satisfied if and only if the corresponding, projected set of constraints involving m-1 variables can be satisfied (see the Appendix).

Given a record with some missing values, we first fill in the observed values into the set of edits. This leads to a set of edits for the variables to be imputed in that record. Next, we sequentially eliminate the variables to be imputed. Each time we eliminate a variable, we obtain an updated set of edits for the remaining variables to be imputed in that record. The main property of Fourier-Motzkin elimination says that if and only if the updated set of edits can be satisfied by the remaining variables, the set of edits we started with can be satisfied by the remaining variables plus the variable we have eliminated (see the Appendix). This property of Fourier-Motzkin elimination allows us to impute the variables in a certain record in a sequential manner by applying the following procedure:

1. Order the variables to be imputed in this record. For simplicity say that the first p variables have to be imputed, and are ordered from x_1 to x_p.
2. Fill in the values of the observed values into the edits. This results in a set of edits T_0 for x_1 to x_p for this record.
3. Eliminate all variables to be imputed except x_p. After we have eliminated the i-th variable ($i = 1,\ldots,p$-1) we obtain the set of edits T_i for the remaining variables, i.e. i+1,…,p. Note that $T_{p\text{-}1}$ defines a feasible interval for the p-th variable.
4. Set $k := p$.
5. If the feasible interval for x_k is degenerate, i.e. contains only one value, we impute this value. Otherwise, we impute the k-th variable using one of our univariate hot deck methods (see Section 3.3 below) so the imputed value lies within the feasible interval for this variable.
6. If $k = 1$, we have imputed all variables in this record and the procedure for this variable terminates. Otherwise continue with Step 7.
7. Fill in the values for x_k to x_p into $T_{k\text{-}2}$. This yields a feasible interval for the $(k-1)$-th variable. Update $k := k-1$, and return to Step 5.

We illustrate the procedure with a simple example below.

Example: Suppose there are four variables, T (turnover), P (profit), C (costs), and N (number of employees), and that the edits are given by (2.2), (2.3),

$$P \leq 0.5T\,, \tag{3.1}$$

$$-0.1T \leq P\,, \tag{3.2}$$

$$T \leq 550N\,. \tag{3.3}$$

Suppose furthermore that in a certain record $N = 5$, and the values of T, P and C are missing. We order the variable as P, C and T. We fill in the value for N into the edits and obtain (2.2), (2.3), (3.1), (3.2) and

$$T \leq 2750\,. \tag{3.4}$$

If we eliminate variable *P*, we use equation (2.2) to express *P* in terms of *T* and *C*. That is, we use $P = T - C$. After Fourier-Motzkin elimination, we obtain the edits (2.3), (3.4),

$$T - C \leq 0.5T\,, \qquad \text{(equivalently: } 0.5T \leq C\text{)} \tag{3.5}$$

and

$$-0.1T \leq T - C \qquad \text{(equivalently: } C \leq 1.1T\text{)} \tag{3.6}$$

The main property of Fourier-Motzkin elimination says that the set of edits (2.3), and (3.4) to (3.6) for *T* and *C* can be satisfied if and only if the original set of edits (2.2), (2.3), and (3.1) to (3.3) for *T*, *P*, *C* and *N* can be satisfied.

If we eliminate variable *C* from edits (2.3), and (3.4) to (3.6), we first copy the edits not involving *C*, i.e. edits (2.3) and (3.4). Moreover, we can eliminate *C* from the edits (3.5) and (3.6), and obtain

$$0.5T \leq 1.1T\,,$$

which is equivalent to (2.3). So, eliminating *C* from (2.3) and (3.4) to (3.6) leads to edits (2.3) and (3.4). The main property of Fourier-Motzkin elimination says that the set of edits (2.3) and (3.4) for *T* can be satisfied if and only if the set of edits (2.3), and (3.4) to (3.6) for *T* and *C* can be satisfied.

Now we impute the variables. Say, we impute the value *T* = 1,200 by means of one of our univariate hot deck imputation algorithms (see below). Filling in this value in the constraints (2.3) and (3.4) to (3.6) that have to hold for *C* and *T*, we obtain that *C* has to satisfy

$$600 \leq C \leq 1320\,.$$

Say we impute the value *C* = 1,000 by means of one of our univariate hot deck imputation algorithms. Filling in the imputed values for *C* and *T* in the constraints (2.2), (2.3), (3.1), (3.2) and (3.4) that have to hold for *P*, *C* and *T*, we obtain that *P* has to satisfy

$$-120 \leq P \leq 600\,.$$

The final step is to use one of our univariate hot deck imputation algorithms to impute a value for *P*. Say, we impute *P* = 200. Our imputed record is then given by *N* = 5, *T* = 1,200, *C* = 1,000 and *P* = 200. It can be verified easily that this imputed record indeed satisfies the edits (2.2), (2.3) and (3.1) to (3.3).

3.3 Univariate hot deck imputation methods

In this paper we apply two classes of hot deck imputation methods: nearest-neighbour imputation and random hot deck imputation.

3.3.1 Nearest-neighbour hot deck imputation

Suppose we want to impute a certain variable x in a record r_0. In the nearest-neighbour approach we calculate for each other record r for which the value of x is not missing (records for which the value of the target variable x is missing can obviously not be used as donor records) a distance given by some distance function.

Before we calculate these distance functions, we first scale the values. We denote the value of variable x_i in record r by x_{ri}, and the corresponding scaled value by x_{ri}^*. We determine the scaled value x_{ri}^* by

$$x_{ri}^* = \frac{x_{ri} - m_i}{s_i},$$

where m_i is the median of the observed values for variable x_i and s_i the interquartile distance, i.e. the difference between the value of the 75% percentile of x_i and the 25% percentile of x_i.

In this paper we consider three different distance functions.

$$d_1(\mathbf{x}_{r_0}^*, \mathbf{x}_r^*) = \sum_{i \in E} w_i \,|\, x_{r_0 i}^* - x_{ri} \,| \qquad (3.7)$$

$$d_2(\mathbf{x}_{r_0}^*, \mathbf{x}_r^*) = \sqrt{\sum_{i \in E} \gamma_i (x_{r_0 i}^* - x_{ri}^*)^2} \qquad (3.8)$$

and

$$d_3(\mathbf{x}_{r_0}^*, \mathbf{x}_r^*) = \max_{i \in E} \gamma_i \,|\, x_{r_0 i}^* - x_{ri}^* \,|, \qquad (3.9)$$

where w_i and γ_i are weights indicating how serious one considers a change of one unit in variable x_i to be, $\mathbf{x}_{r_0}^*$ is the scaled recipient record and $\mathbf{x}_r^*$ a scaled potential donor record. In this paper we have set $\gamma_i = 1$ for all variables (i=1,…,n). E is the set of observed variables in the recipient record $\mathbf{x}_{r_0}$. With $\max_{i \in E}$ we indicate that the maximum over all variables in E is taken.

If one or more values for variables in E of the donor record $\mathbf{x}_r$ is missing, we set the values of these variables equal to zero in both the donor record and the recipient record while calculating (3.7) to (3.9). Note that this way of dealing with missing values in donor records deviates from the "conventional" way of dealing with such missing values, where donor pools are formed based on auxiliary variables that are observed for donors and recipients (see, e.g., Andridge and Little, 2010). The conventional way would be to consider only donors where all so-called "matching variables", i.e. variables that one wants to use to match potential donor records to the recipient record, are observed. These records form a "donor pool" from which a donor record for the recipient record is subsequently drawn. If a donor pool is empty, some of the matching variables have to be deleted in order to obtain a non-empty donor pool. For data sets containing a large number of missing values as in our evaluation data sets, constructing donor pools for each recipient can become quite cumbersome. As the main aim of our paper is to illustrate how the class of hot deck imputation techniques can be extended so that edits are satisfied rather than developing "optimal" imputation methods, we have chosen a simpler way to deal with missing values in donor records.

To impute a missing value, we first select the potential donor value from the record with the smallest distance. If the value is allowed according to the edits, we use it to impute the missing values. If that value is not allowed according to the edits, we try the potential donor value from the record second smallest distance, et cetera until we find a donor value that is allowed according to the edits. To limit the computing time we try at most 160 different potential donor values. If none of these 160 potential donor values lies within the feasible interval for the variable to be imputed, we impute the value on the boundary of the feasible interval that is closest to the first potential donor value.

As a remark, we note that if we used the subset of variables that are observed for all records in (3.7), (3.8) or (3.9) 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 were to lead to non-preserved totals, a value from another potential donor record would be used.

3.3.2 Random hot deck imputation

In our application of random hot deck imputation, we construct an ordered list of potential donor records for each record with missing values by randomly drawing (without replacement) potential donor records, until all potential donor records have been drawn and put on the list for this recipient record.

To impute a missing value in a certain recipient record, we select the first donor record on the ordered list of donors for this recipient for which the corresponding value is observed. We then check whether that value lies in the feasible interval for the missing value under consideration. If so, we use it to impute this missing value. Otherwise, we select the next donor record on the ordered list of donors for this recipient for which the corresponding value is observed, and check whether that value lies in the feasible interval for the missing value under consideration, et cetera until we find a donor value that does lie in the feasible interval for the missing value under consideration. If none of the possible donor values lies in the feasible interval for the variable to be imputed, we impute the value of boundary of the feasible interval that is closest to the first potential donor value on the ordered list.

Note that instead of ordering the potential donors of each variable separately, we could alternatively have constructed an ordered list of potential donor values for the first variable to be imputed only and use that ordered for all subsequent variables to be imputed as well. In that case, if possible, multivariate imputation, using several values from the first potential donor record on this list, would be used.

One would expect that random hot deck imputation preserves individual values less well than nearest-neighbour imputation. On the other hand, random hot deck imputation may preserve the statistical distribution better than nearest-neighbour imputation. In Section 4 we examine whether these expectations are fulfilled.

4. Evaluation study

4.1 Evaluation data

For our evaluation study we have used three data sets: a data set with actually observed data from a business survey, data set R^{all}, the same data set but without balance edits, data set R^{ineq}, and a data set with synthetic data, data set S. The data sets R^{all} and R^{ineq} contain raising weights. These raising weights differ across different (strata of) records, and are used in some of our evaluation measures. In data S all raising weights were set to 1. The main characteristics of these data sets are presented in Table 1.

Table 1. The characteristics of the evaluation data sets.

	Data set R^{all}	Data set R^{ineq}	Data set S
Total number of records	3,096	3,096	500
Number of records with missing values	544	469	490
Total number of variables	8	7	10
Total number of edits	14	12	16
Number of balance edits	1	0	3
Total number of inequality edits	13	12	13
Number of non-negativity edits	8	7	9

The actual values for data set R^{all}, and hence also for data set R^{ineq}, are all known. In the completely observed data set values were deleted by a third party, using a mechanism unknown to us. Data set R^{ineq} was constructed in order to examine the effects of balance edits on the results. To construct R^{ineq} we have removed the balance edit, one of the variables (R_4) involved in the balance edit, and its associated non-negativity edit from R^{all}. The removed variable R_4 does not occur in any of the other edits apart from its associated non-negativity edit.

Data set S is indirectly based on an observed business survey and its corresponding edits. These observed data were used to estimate the parameters of a multivariate normal model by means of the EM algorithm. Next, data set S was generated by drawing from the estimated multivariate normal model. If a drawn vector did not satisfy all specified edits, it was rejected, else it was accepted. In this way 500 records were generated. Missing values were generated by randomly deleting for each variable a specified number of values. The number of values deleted was (much) higher than in the actually observed business survey in order to evaluate the performance of our imputation approaches for a very complicated situation.

For all three data sets we have two versions available: a version with missing values and a version with complete records. The former version is imputed. The resulting data set is then compared to the version with complete records, which we consider as a data set with the true values.

The numbers of missing values and (unweighted) means of the 8, respectively 7, variables of data set R^{all} and data set R^{ineq} are given in Table 2 and those of the 10 variables of data set S in Table 3. The means are taken over all observations in the complete versions of the data sets.

Table 2. The numbers of missing values and the means of the variables of data sets R^{all} and R^{ineq}.

Variable	Number of missing values	Mean
R_1	76	11,574.83
R_2	79	777.56
R_3	130	8,978.70
R_4*	147	1,034.07
R_5	68	10,012.77
R_6	67	169.24
R_7	73	209.86
R_8	0	37.41

* Data set R^{ineq} does not contain variable R_4.

Variable R_8 does not contain any missing values and is only used as auxiliary variable.

Table 3. The numbers of missing values and the means of the variables of data set S.

Variable	Number of missing values	Mean
S_1	120	97.77
S_2	180	175,018.30
S_3	240	731.03
S_4	120	175,749.33
S_5	180	154,286.53
S_6	180	7,522.34
S_7	180	8,519.65
S_8	180	1,277.04
S_9	120	171,605.57
S_{10}	120	4,143.76

4.2 Evaluation measures

To measure the performance of our imputation approaches we use a d_{L1} measure, an m_1 measure, an *rdm* measure, and Kolmogorov-Smirnov distance. The first two criteria have been proposed by Chambers (2003). The d_{L1} measure is the average distance between the imputed and true values defined as

$$d_{L1} = \frac{\sum_{r \in M} w_r \left| \hat{y}_r - y_r^* \right|}{\sum_{r \in M} w_r},$$

where $\hat{y}_r$ is the imputed value in record *r* of the variable under consideration, *M* denotes the set of records with imputed values for variable *y* and w_r is the raising weight for record *r*.

The m_1 measure, which measures the preservation of the first moment of the empirical distribution of the true values, is defined as

$$m_1 = \left| \sum_{r \in M} w_r (\hat{y}_r - y_r^*) \Big/ \sum_{r \in M} w_r \right|.$$

The *rdm* (relative difference in means) measure has been used in an evaluation study by Pannekoek and De Waal (2005), and is defined as

$$rdm = \frac{\sum_{r \in M} \hat{y}_r - \sum_{r \in M} y_r^*}{\sum_{r \in M} y_i^*}.$$

To remain consistent with the literature, in particular with the previously published papers by Chambers (2003) and Pannekoek and De Waal (2005), we have not made an attempt to make the d_{L1} and the m_1 measures comparable across variables.

Finally, we use the *K-S* Kolmogorov-Smirnov distance to compare the empirical distribution of the original values to the empirical distribution of the imputed values (also proposed by Chambers, 2003). For unweighted data, the empirical distribution of the original values is defined as: $F_{x^*}(t) = \sum_{r \in M} I(x_r^* \leq t) / m$, with m the number of records with missing values for the record at hand (i.e. the size of set M), and similarly $F_{\hat{x}}(t)$ where I is the indicator function. The *K-S* distance is defined as

$$K\text{-}S = \max_j (| F_{x^*}(t_j) - F_{\hat{x}}(t_j) |),$$

where the t_j values are the $2p$ jointly ordered original and imputed values of x with p the number of missing values of x.

Smaller absolute values of the evaluation measures indicate better imputation performance.

4.3 Evaluation results

The results for data set R$^{\text{all}}$ are presented in Tables 4 and 5. In these tables, as well as in following tables, “A” (from “Absolute”) refers to the imputation method that uses distance function (3.7), “E” (from “Euclidean”) to the method that uses distance function (3.8), “M” (from “Maximum”) to the method that uses distance function (3.9), and “R” (from “Random”) to the method that draws donors randomly.

Table 4. Evaluation results for d_{L1} *and* m_1 *for data set* R^{all}

Variable	d_{L1}				m_1			
	A	E	M	R	A	E	M	R
R_1	19134	1769	1924	12091	1756	1377	1902	10582
R_2	111	108	116	195	68	54	89	10
R_3	489	476	526	497	485	465	524	486
R_4	453	440	500	1157	448	428	499	1151
R_5	29	30	35	1512	22	23	8	1489
R_6	3.09	1.45	3.26	1.38	0.61	1.10	0.85	1.16
R_7	34	33	14	39	21	20	0	29

Table 5. Evaluation results for rdm and K-S for data set R^{all}

Variable	*rdm*				*K-S*			
	A	E	M	R	A	E	M	R
R_1	-0.65	-0.33	-0.94	0.99	0.47	0.34	0.72	0.16
R_2	-0.39	-0.31	-0.59	-0.48	0.23	0.23	0.29	0.22
R_3	-0.36	-0.35	-0.39	-0.36	0.22	0.16	0.31	0.20
R_4	3.12	3.03	3.49	3.78	0.18	0.03	0.05	0.03
R_5	-0.03	-0.03	-0.01	0.08	0.03	0.03	0.03	0.03
R_6	-0.81	-1.00	-0.73	-0.98	0.01	0.07	0.04	0.04
R_7	3.94	1.79	-0.13	0.24	0.12	0.15	0.11	0.14

Variable R_8 does not have any missing values, so no evaluation results for R_8 are presented.

The results for data set R^{ineq} are presented in Tables 6 and 7.

Table 6. Evaluation results for d_{L1} and m_1 for data set R^{ineq}

Variable	d_{L1}				m_1			
	A	E	M	R	A	E	M	R
R_1	1924	1606	12387	18239	1922	1597	10753	16140
R_2	116	101	160	198	71	68	160	5
R_3	1430	1551	1518	28949	1424	1091	1391	27010
R_5	1379	1317	1670	6120	1352	981	1571	3874
R_6	2.18	2.16	1.40	3.30	0.36	0.38	0.92	0.76
R_7	17	19	18	42	2	4	1	28

Table 7. Evaluation results for rdm and K-S for data set R^{ineq}

Variable	*rdm*				*K-S*			
	A	E	M	R	A	E	M	R
R_1	-0.84	-0.66	2.02	1.27	0.49	0.37	0.62	0.13
R_2	-0.45	-0.48	-0.91	-0.45	0.22	0.20	0.75	0.24
R_3	-0.98	-0.62	-0.97	1.00	0.61	0.52	0.72	0.08
R_5	-0.77	-0.49	-0.97	-0.31	0.35	0.33	0.60	0.19
R_6	-0.95	-0.95	-0.85	-0.95	0.03	0.04	0.06	0.04
R_7	-0.29	-0.24	-0.39	-0.46	0.14	0.12	0.21	0.16

The results for data set S are presented in Tables 8 and 9.

Table 8. Evaluation results for d_{L1} and m_1 for data set S

Variable	d_{L1}				m_1			
	A	E	M	R	A	E	M	R
S_1	40	38	33	34	19	20	9	15
S_2	15574	17593	24225	15759	736	1077	7451	327
S_3	9971	13099	15016	12139	9929	13058	14970	12106
S_4	21869	26504	43687	24312	20962	24501	41117	23722
S_5	54985	67707	54833	48129	50619	61510	44347	48129
S_6	2441	2326	2153	2492	1707	1340	1516	2011
S_7	19942	23358	23198	21983	16900	20103	20053	18794
S_8	37887	46147	39882	32659	37807	46070	39882	32659
S_9	16684	19303	36004	22924	3571	4984	20923	9888
S_{10}	21452	25099	34182	24505	17391	19517	20194	13833

Table 9. Evaluation results for rdm and K-S for data set S

Variable	*rdm*				*K-S*			
	A	E	M	R	A	E	M	R
S_1	0.19	0.21	0.09	0.15	0.21	0.22	0.15	0.17
S_2	0.00	0.01	0.04	0.00	0.06	0.04	0.06	0.04
S_3	13.36	17.57	20.14	16.29	0.85	0.86	0.87	0.87
S_4	0.12	0.15	0.24	0.14	0.12	0.09	0.16	0.11
S_5	-0.34	-0.41	-0.30	-0.29	0.28	0.36	0.26	0.23
S_6	-0.22	-0.17	-0.19	-0.26	0.19	0.13	0.09	0.17
S_7	1.90	2.26	2.25	2.11	0.20	0.19	0.17	0.19
S_8	28.49	34.72	29.96	24.57	0.53	0.52	0.51	0.52
S_9	0.02	0.03	0.12	0.06	0.04	0.03	0.09	0.07
S_{10}	4.48	5.02	5.20	3.56	0.23	0.23	0.23	0.23

The first conclusion that can be drawn from the Tables 4 to 9 is a rather disappointing one, namely that no method stands out as the best across all evaluation measures and all data sets. Neither does one method stand out as the worst across all evaluation measures and all data sets.

Our evaluation measures can be divided into an evaluation measure that measures the preservation of individual values (d_{L1}), two evaluation measures that measure the preservation of means (m_1 and *rdm*), and one evaluation measure that measures the preservation of the statistical distribution (*K-S*).

Examining the preservation of the individual values by means of the d_{L1} measure for all variables in all data sets we see that methods “A” and “E” are best in 9, respectively 7, cases (where a case is a combination of evaluation measure and variable in any of the data sets), and worst in only 2, respectively 3, cases. For the d_{L1} measure methods “M” and “R” are best in only 2, respectively 3, cases, and worst in 7, respectively 11, cases. For the d_{L1} measure we therefore conclude that methods “A” and “E” perform best for these data sets.

Examining the preservation of means using the m_1 and *rdm* measures we see that the methods do not differ much with respect to how often they are best. Method “Q” is best in 15 cases, followed by method “A”(12 cases), method “R” (11 cases) and finally method “M” (10 cases). The methods do differ quite substantially from each other with respect to in how many cases they are the worst: method “A” is worst is only 2 cases, “E” in 10 cases, “R” in 17 cases and “M” even in 19 cases. For the preservation of means we again conclude that methods “A” and “E” perform best for these data sets.

Finally, examining the preservation of the statistical distribution using the *K-S* measure, we see a different picture. Here method “R” is best most often, namely in 10 cases versus 9 cases for method “E”, 7 cases for method “M” and 5 cases for method “A”. Moreover, method “R” is worst in only 1 case versus 4 cases for method “E”, 5 cases for method “A”, and 7 cases for method “M”. For the preservation of the statistical distribution we hence conclude that method “E” performs best for the data sets used in our evaluation study.

Carrying out a similar analysis for data sets $\mathrm{R}^{\mathrm{all}}$ and $\mathrm{R}^{\mathrm{ineq}}$ separately, we see that method “E” is best for evaluation measures d_{L1}, m_1 and *rdm* for both data sets. Method “R” is worst for evaluation measures d_{L1} and m_1 for both data sets. For evaluation measure *rdm* method “R” is worst for data set $\mathrm{R}^{\mathrm{all}}$ and second worst for data set $\mathrm{R}^{\mathrm{ineq}}$. Worst for data set $\mathrm{R}^{\mathrm{ineq}}$ with respect to this evaluation measure is method “M”. Finally, for evaluation measure *K-S* method “R” is best, and method “M” worst. Our conclusion is that the exclusion of the balance edit from data set $\mathrm{R}^{\mathrm{all}}$ hardly affects the evaluation results.

The situation is different when we compare the evaluation results for data sets $\mathrm{R}^{\mathrm{all}}$ and $\mathrm{R}^{\mathrm{ineq}}$ on the one hand and data set S on the other. If we look at the evaluation results for both $\mathrm{R}^{\mathrm{all}}$ and $\mathrm{R}^{\mathrm{ineq}}$, we can conclude that methods “E” and “M” are best, respectively worst, for evaluation measures d_{L1}, m_1 and *rdm*. For data set S, however, method “A” performs best for evaluation measures d_{L1}, m_1 and *rdm*. The worst methods for these evaluation measures are methods “E” and “M”. For method “M” this is not very surprising as this was also the worst method with respect to these evaluation measures for data set R^{all} and R^{ineq}. For method "E", however, this is rather surprising as this was the best method for those data sets. For evaluation measure *K-S* methods "R" and "E" can be considered to be best, and method "A" worst.

An important conclusion of the above analysis is that with respect to preservation of individual values and preservation of means, method "R" is the worst method for the examined data sets, whereas it is the best method with respect to the preservation of the statistical distribution. This confirms what is considered as conventional wisdom:

> *by means of imputation one can either try to preserve individual values and/or means or one can try to preserve the statistical distribution, but preserving all these aspects at the same time appears to be too complicated in general.*

Some of the evaluation results for evaluation measure *rdm* are remarkably large, for instance the value 20.14 for variable S_3 and method "M", indicating that the imputed values for this variable are on average more than 20 times as large as the true values. Unfortunately, this is not caused by a programming error or a mistake during the evaluation. Instead these large errors point to a drawback of (our implementation of) the developed imputation methods. The average value of variable S_3 is relatively small in comparison to most other variables, as can be seen in Table 3. However, variable S_3 is involved in a balance edit, involving other – on average much larger – variables. As variable S_3 also has many missing values it is imputed at the end of the imputation process in our implementation. It is hence essentially computed as the difference between the (possibly imputed) total and the sum of the (possibly imputed) other constituent variables in the balance edit. An imputation error, i.e. the difference between the imputed value and the true value, in any of the other imputed variables may be relatively small for that variable but may be very large for variable S_3.

In Coutinho, De Waal and Remmerswaal (2011) a related imputation method that also satisfies edits has been examined. In that article the (potential) imputation values were, however, generated by means of a posited multivariate normal model rather than by means of the hot deck methods of the present paper. In the present paper we will denote that method as "N", referring to the normal model. In the same article the use of a multivariate normal model to generate pre-imputation values without taking the edits into account was also examined. Later these pre-imputation values were adjusted so that the final imputation values satisfy all edits. In the present paper we will refer to this method as the adjustment method. This method is also used in the software package SLICE (see De Waal, 2001). In the evaluation study in Coutinho, De Waal and Remmerswaal (2011) the *K-S* measure was not used. To compare the results of the 6 methods we can hence only use evaluation measures d_{L1}, m_1 and *rdm*.

The results of method "N" and the adjustment method are given in Tables 10 to 12. To limit the computing time for method "N" at most 160 different potential donor values were generated by the multivariate normal model. If none of these 160 potential donor values lay within the feasible interval for the variable to be imputed, we imputed the value on the boundary of the feasible interval that was closest to the first potential donor value.

Table 10. Evaluation results for the adjustment method and method N for data set R^{all}

Variable	d_{L1}		m_1		*rdm*	
	N	Adjustment	N	Adjustment	N	Adjustment
R_1	3141.27	2069.20	2593.45	1145.80	0.34	0.15
R_2	277.30	226.91	222.28	108.27	0.34	0.17
R_3	176.55	158.79	142.97	106.63	-0.05	-0.04
R_4	189.06	532.81	160.18	531.39	0.44	3.58
R_5	65.59	14.81	54.20	14.81	0.00	-0.01
R_6	13.59	41.00	13.17	40.617	0.90	2.65
R_7	83.83	86.37	80.40	75.14	1.77	1.42

Table 11. Evaluation results for the adjustment method and method N for data set R^{ineq}

Variable	d_{L1}		m_1		*rdm*	
	N	Adjustment	N	Adjustment	N	Adjustment
R_1	3101.88	1868.20	2717.77	256.14	0.33	-0.26
R_2	271.09	205.16	216.54	34.67	0.29	-0.38
R_3	360.10	1490.70	279.12	1452.00	-0.09	-0.99
R_5	1837.67	1227.90	1756.52	541.04	0.14	-0.49
R_6	13.77	2783.80	13.37	2783.80	0.84	592.50
R_7	92.65	14.40	89.43	12.03	1.93	-0.54

Table 12. Evaluation results for the adjustment method and method N for data set S

Variable	d_{L1}		m_1		*rdm*	
	N	Adjustment	N	Adjustment	N	Adjustment
S_1	13943.12	466.17	13916.90	452.17	142.57	4.63
S_2	17440.92	44304.04	8066.39	42833.46	0.05	-0.26
S_3	9941.38	32441.33	9767.14	32332.18	13.14	43.50
S_4	32672.09	28114.87	31633.86	673.03	0.19	0.00
S_5	11404.99	56780.58	5274.79	49973.13	-0.04	-0.33
S_6	2221.02	33203.49	1430.56	28916.08	0.18	3.72
S_7	3472.59	22135.07	1405.63	15792.76	0.16	1.77
S_8	5062.49	75627.02	4818.50	75118.57	3.63	56.62
S_9	5715.68	127862.42	3569.85	104736.50	0.02	0.60
S_{10}	28261.21	145238.77	28064.01	104194.00	7.22	-26.82

We again carry out a similar analysis as before, but this time we compare all 6 methods for evaluation measures d_{L1}, m_1 and *rdm* simultaneously. For evaluation measures d_{L1}, m_1 and *rdm* and data sets R^{all} and R^{ineq} we can conclude that the adjustment method and method “E” are slightly preferable over methods “N”, “A” and “M”. Clearly worst is method “R” for these data sets. For data set S, method “N” turns out to be clearly best, followed by method “A”. The adjustment method performed worst on this data set. Overall method “N” appears to perform best, followed by method “E”.

5. Discussion

In this paper we have extended hot deck imputation methods so that the imputed data satisfy specified edits. The hot deck imputation methods we have considered are random hot deck imputation and nearest neighbour hot deck imputation based on 3 different distance functions to measure the distance between the record to be imputed and possible donor records. To ensure that the edits become satisfied after imputation, we have applied these hot deck imputation methods in a sequential manner and have used Fourier-Motzkin elimination to determine feasible intervals for each variable to be imputed. In our extended versions of these imputation methods we aim to use the same donor record for all missing values in a record to be imputed, unless the edits dictate that this is not possible. In the latter case, we try to select additional donor records for this recipient record. If even this turns out to be impossible due to the edits, we set (some of) the missing values to values that ensure edits to be satisfied.

In Coutinho, De Waal and Remmerswaal (2011) we have developed an imputation method – referred to as method "N" in Section 4 – based on a posited multivariate normal model that also satisfies specified edits. The practical advantage of the imputation methods developed in the present paper over the method developed in Coutinho, De Waal and Remmerswaal (2011) is that they are much easier to implement in a computer programme. From a practical point of view this may make the methods developed in this paper more attractive for potential users, especially since the evaluation results for method "N" only slightly prevail over the hot deck imputation methods.

Some potential users may consider even the methods described in the current paper too complex or too costly to implement. For those potential users, the adjustment approach as described in Coutinho, De Waal and Remmerswaal (2011) seems a good option. That method is easy to implement and use in practice. The price that has to be paid is that this method performs slightly worse than the hot deck imputation methods examined in this paper.

We have applied the developed imputation methods to 3 data sets, 2 of which differed only slightly. The third data set differed substantially from the other two. The main conclusion we can draw from our evaluation study is that it depends on the specific data set and the specific aims of imputation which of the developed imputation method is best. Even if one pursues the same aim, different imputation methods can be best for different data sets. If the aim of imputation is to preserve the statistical distribution of the imputed data as well as possible, random hot deck imputation performed better than the nearest neighbour imputation methods in our evaluation study.

A drawback of the sequential imputation methods developed in the current paper, and in Coutinho, De Waal and Remmerswaal (2011), is that the imputation error for a variable that is relatively small and is involved in a balance edit, involving other – on average much larger – variables may be very large in comparison to its value. This may happen in particular if such a variable is one of the last variables to be imputed.

Perhaps changing the order in which the variables are imputed may help to overcome the above problem. In general, the best order for imputing the variables for sequential imputation methods such as the ones we have developed in our paper is an open problem at the moment.

As a simple diagnostic measure to detect such large imputation errors, we propose to compare the average imputed value for each variable with the average value of the observed data for this variable. A large difference points to potential problems that need to be examined in more detail. Undoubtedly, more and more appropriate diagnostic measures can be developed.

Of course, from a qualitative point of view, even better than trying to improve sequential imputation methods might be to develop an imputation method that imputes all missing values in a record simultaneously based on their (estimated) joint statistical distribution. In principle, with such a simultaneous imputation method one would be able to overcome the basic drawback of a sequential imputation method that optimal choices for individual variables to be imputed may not lead to overall optimality for all variables. However, such simultaneous imputation method that satisfy all edits seem exceedingly hard to develop. Even for the “simplest” case where the data are assumed to follow a truncated multivariate normal distribution, such imputations become theoretically very complicated and computationally very demanding as illustrated by Tempelman (2009) and De Waal, Pannekoek and Scholtus (2011; Chapter 9). Further research is required to develop such simultaneous imputation methods that are computationally tractable in practical situations.

References

Andridge, R.A. and R.J.A. Little (2010), A Review of Hot Deck Imputation for Survey Non-response. *International Statistical Review 78*, pp. 40-64.

Chambers, R. (2003), Evaluation Criteria for Statistical Editing and Imputation. In: *Methods and Experimental Results from the EUREDIT Project* (ed. J.R.H. Charlton) (available on http://www.cs.york.uk/euredit/).

Coutinho, W., T. de Waal and M. Remmerswaal (2011), Imputation of Numerical Data under Linear Edit Restrictions. *SORT 35*, pp. 39-62.

De Waal, T. (2001), SLICE: Generalised Software for Statistical Data Editing. *Proceedings in Computational Statistics* (ed. J.G. Bethlehem and P.G.M. Van der Heijden), Physica-Verlag, New York, pp. 277-282.

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

Duffin, R.J. (1974), On Fourier’s Analysis of Linear Inequality Systems. *Mathematical Programming Studies 1*, pp. 71-95.

Geweke, J. (1991), *Efficient Simulation from the Multivariate Normal and Student-t Distributions Subject to Linear Constraints and the Evaluation of Constraint Probabilities*. Report, University of Minnesota.

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

Kovar, J. and P. Whitridge (1990), Generalized Edit and Imputation System; Overview and Applications. *Revista Brasileira de Estadistica 51*, pp. 85-100.

Kovar, J. and P. Whitridge (1995), Imputation of Business Survey Data. In: *Business Survey Methods* (ed. Cox, Binder, Chinnappa, Christianson & Kott), John Wiley & Sons, New York, pp. 403-423.

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

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

Pannekoek, J. and T. De Waal (2005), Automatic Edit and Imputation for Business Surveys: the Dutch Contribution to the EUREDIT Project. *Journal of Official Statistics 21*, pp. 257-286.

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

Särndal, C.-E. and S. Lundström (2005), *Estimation in Surveys with Nonresponse*. John Wiley & Sons, Chichester.

Schafer, J.L. (1997), *Analysis of Incomplete Multivariate Data*. Chapman & Hall, London.

Tempelman, C. (2007), *Imputation of Restricted Data*. Doctorate thesis, University of Groningen.

Winkler, W.E. and L.A. Draper (1997), The SPEER Edit System. *Statistical Data Editing (Volume 2); Methods and Techniques*, United Nations, Geneva.

Appendix: Fourier-Motzkin elimination

The standard version of Fourier-Motzkin elimination handles only inequalities as constraints. We use an extended version of Fourier-Motzkin elimination that can also handle equations. In our application of Fourier-Motzkin elimination the constraints are defined by the edits for a record. We assume there *m* variables and a set of (in)equalities given by (2.1).

In order to eliminate a variable x_q from the set of current edits by means of Fourier-Motzkin elimination, we start by copying all edits not involving this variable from the set of current edits to a new set of edits Ψ.

If variable x_q occurs in an equation, we express x_q in terms of the other variables. Say, x_q occurs in edit *s* of type (2.1a), we then write x_q as

$$x_q = -\frac{1}{a_{qs}}\left(b_s + \sum_{i \neq q} a_{is} x_i\right) \tag{A.1}$$

Expression (A.1) is used to eliminate x_q from the other edits involving x_q. These other edits are hereby transformed into new edits, not involving x_q, that are logically implied by the old ones. These new edits are added to our new set of edits Ψ. Note that if the original edits are consistent, i.e. can be satisfied by certain values u_i (*i*=1,…,*m*), then the new edits are also consistent as they can be satisfied by u_i (*i*=1,…,*m*; $i \neq q$). Conversely, note that if the new edits are consistent, say they can be satisfied by the values v_i (*i*=1,…,*m*; $i \neq q$), then the original edits are also consistent as they can be satisfied by the values v_i (*i*=1,…,*m*) where v_q is defined by filling v_i (*i*=1,…,*m*; $i \neq q$) into (A.1).

If x_q does not occur in an equality but only in inequalities, we consider all pairs of edits (2.1b) involving x_q. Suppose we consider the pair consisting of edit *s* and edit *t*. We first check whether the coefficients of x_q in those inequalities have opposite signs, i.e. we check whether $a_{qs} \times a_{qt} < 0$. If this is not the case, we do not consider this particular combination (*s*,*t*) anymore. If the coefficients of x_q do have opposite signs, one of the edits, say edit *s*, can be written as an upper bound on x_q, i.e. as

$$x_q \leq -\frac{1}{a_{qs}}\left(b_s + \sum_{i \neq q} a_{is} x_i\right), \tag{A.2}$$

and the other edit, edit *t*, as a lower bound on x_q, i.e. as

$$x_q \geq -\frac{1}{a_{qt}}\left(b_t + \sum_{i \neq q} a_{it} x_i\right). \tag{A.3}$$

Edits (A.2) and (A.3) can be combined into

$$-\frac{1}{a_{qt}}\left(b_t + \sum_{i \neq q} a_{it} x_i\right) \leq x_q \leq -\frac{1}{a_{qs}}\left(b_s + \sum_{i \neq q} a_{is} x_i\right),$$

which yields an implied edit not involving x_q given by

$$-\frac{1}{a_{qt}}\left(b_t + \sum_{i \neq q} a_{it} x_i\right) \leq -\frac{1}{a_{qs}}\left(b_s + \sum_{i \neq q} a_{is} x_i\right). \tag{A.4}$$

The implied edit (A.4) is added to our new set of edits Ψ. After all possible pairs of edits involving x_q have been considered and all implied edits given by (A.4) have been generated and added to Ψ, we delete the original edits involving x_q that we started with. In this way we obtain a new set of edits Ψ not involving variable x_q. This set of edits Ψ may be empty. This occurs when all current edits involving x_q are inequalities and the coefficients of x_q in all those inequalities have the same sign. Note that if the original edits are consistent, say they can be satisfied by certain values u_i (*i*=1,…,*m*), then the new edits are also consistent as they can be satisfied by u_i (*i*=1,…,*m*; $i \neq q$). This is by definition also true if the new set of edits is empty. Conversely, note that if the new edits are consistent, say they can be satisfied by certain values v_i (*i*=1,…,*m*; $i \neq q$), then the minimum of the right-hand sides of (A.4) for the v_i (*i*=1,…,*m*; $i \neq q$) is larger than, or equal to, the maximum of the left-hand sides of (A.4) for the v_i (*i*=1,…,*m*; $i \neq q$). This implies that we can find a value v_q such that

$$-\frac{1}{a_{qt}}\left(b_t + \sum_{i \neq q} a_{it} v_i\right) \leq v_q \leq -\frac{1}{a_{qs}}\left(b_s + \sum_{i \neq q} a_{is} v_i\right) \text{ for all pairs } s \text{ and } t,$$

which in turn implies that the original edits are consistent. We have demonstrated the main property of Fourier-Motzkin elimination: a set of edits is consistent if and only if the set of edits after elimination of a variable is consistent.