Data editing

Detection and correction of errors

10

Jeffrey Hoogland, Mark van der Loo, Jeroen Pannekoek and Sander Scholtus

Statistical Methods (201110)

Statistics Netherlands

The Hague/Heerlen

Explanation of symbols

.	= data not available
*	= provisional figure
**	= revised provisional figure
x	= publication prohibited (confidential figure)
–	= nil or less than half of unit concerned
–	= (between two figures) inclusive
0 (0,0)	= less than half of unit concerned
blank	= not applicable
2010–2011	= 2010 to 2011 inclusive
2010/2011	= average of 2010 up to and including 2011
2010/'11	= crop year, financial year, school year etc. beginning in 2010 and ending in 2011
2008/'09–2010/'11	= crop year, financial year, etc. 2008/'09 to 2010/'11 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

60165201110 X-37

1. Introduction to the theme

1.1 General description and reading guide

1.1.1 Description of detection and correction

Errors are virtually always present in the data files used at Statistics Netherlands. This is true for both the data obtained from Statistics Netherlands' own observations and for data originating from external registers. Insofar as these errors result in bias in publication figure estimates, it is important for Statistics Netherlands to detect and correct these errors.

Errors can arise during the observation; if this is the case, there will be a difference between the reported value and the actual value. This can occur because the respondent does not know the actual value exactly or at all, or has difficulty finding this value and therefore makes an estimate. Another possible cause is the difference in definitions between the accounting records of companies and Statistics Netherlands, because, for example, the financial year differs from the calendar year. Furthermore, it is possible that companies simply do not measure information that Statistics Netherlands wants to receive. In this case, the respondent will again estimate the value or not fill it in at all. Finally, respondents may also read or understand questions incorrectly. For example, they may report in euros, while they were actually asked to report in thousands of euros, or a respondent may answer only for him/herself and not, as asked, for the entire household.

Errors can also arise during the procedure for processing the data after it has been collected. At Statistics Netherlands, the collected data goes through different processes, such as entering, coding, detection, correction, weighting and tabulation. All of these processes can introduce errors into the data. An example of this is that the manual entry of data can result in misinterpretations, for example, a '1' is taken for a '7' or vice versa. Additionally, there may be errors in the processing software, or good values may incorrectly be seen as errors during the detection and correction process.

Detection and correction methods have various objectives:

1. To identify possible sources of errors so that the statistical process can be improved in the future;
2. To provide information about the quality of the data collected and published;
3. To detect and correct influential errors in the data collected;
4. To provide complete and consistent data.

Currently at Statistics Netherlands, detection and correction methods are used primarily to provide complete and consistent data and to correct errors that have a

significant influence on the publication total. In addition, based on the errors found, improvements are made to the layout of the questionnaire or additional explanations are requested. An analysis of the errors found can also be used to establish differences between electronic and written questions and to obtain insight into the quality of the administrative data.

Different detection and correction methods and processes have been developed for different types of errors. What is important here is the distinction between influential errors and non-influential errors and the distinction between systematic and random errors.

A distinction can be made between *influential* and *non-influential errors* particularly in business statistics. Influential errors include the errors that have a significant influence on the final publication total. This can arise because the error was made by a large company that already has a significant influence on the total, or by a smaller company that is weighted heavily in the estimate for the total, or because a large error was made that will strongly influence the total, such as a thousand-error. It is clear that errors that have a large influence on a publication total can lead to significant bias; these are also very high-risk for Statistics Netherlands. For this reason, it is crucial to detect and correct these errors as effectively as possible. The detection and correction process will also have to mainly focus on these errors.

Another breakdown that is often made is that between *systematic* and *random errors*. A systematic error is an error that is made by multiple respondents, such as the abovementioned thousand-errors, or the fact that different respondents stated their gross instead of net income data, or the fact that a group of respondents placed a minus sign before a figure, while the minus sign was already included on the questionnaire. Because these errors are made by multiple respondents in the same way, they can produce a systematic bias. If it is known what systematic errors were made, then these are often easy to detect and correct. Random errors are errors that arise by accident. The most common cause of this is inattentiveness on the part of the respondent, the interviewer or the person entering the data. An example of this is an error made because two figures were interchanged when writing down the answer. When these errors are made non-systematically, the risk of a systematic bias arising due to this type of error will be smaller.

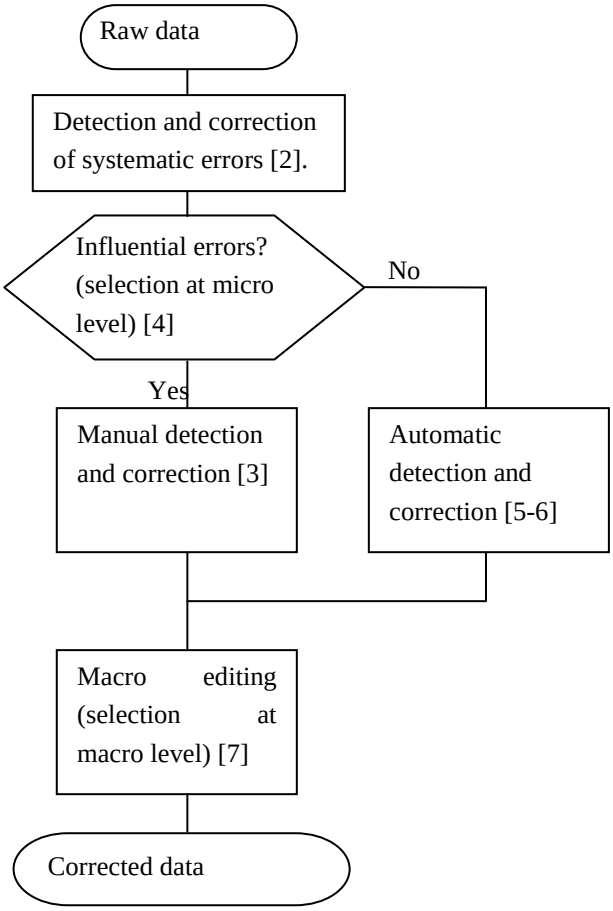
Systematic errors can be both influential (thousand-error) and non-influential (sign error in a small value). The same applies for random errors. If, for example, a large company fills in too many figures accidentally, this can be influential. However, if a smaller company does this, the error could possibly be non-influential.

In this theme, we discuss detection and correction methods developed to detect and correct systematic, random and influential errors. In Section 1.1.2, we will describe the different types of methods using a prototype process. We will set out in a general manner the different process steps of a possible detection and correction process.

1.1.2 Problems and solutions

The detection and correction process starts after the data has been collected and entered. The specific way that this process is used will vary by statistic. However, there is a general strategy that will be followed in broad lines in many processes. This general strategy is shown in Figure 1 and provides an overview of the detection and correction process.

Figure 1. Overview of the detection and correction process[1)]



[1)] *The figures between square brackets refer to the related chapters in this report.*

In the first phase of the detection and correction process, identifiable systematic errors are detected and corrected. As stated previously, these systematic errors can lead to significant bias. Moreover, these errors can often be automatically detected and corrected easily and very reliably. It is highly efficient to correct these errors at an early stage. The detection and correction of systematic errors is discussed in Chapter 2 of this report.

After the identifiable systematic errors have been corrected automatically, a decision can be taken to begin manual detection and correction. This process step is performed by editors or analysts who are usually supported in this regard by

software that allows, for example, edit rules to be applied and values to be changed interactively. We therefore refer to both *manual editing* and *interactive editing*. This form of editing is described in Chapter 3 of this report.

Manual detection and correction is expensive and time consuming. It is therefore better to examine only records with influential errors manually so that the specialists' limited time can be used where it is most effective. This means that the records that are expected to contain influential errors are selected for interactive editing. The other records with less important errors can be edited automatically. If the automatic editing is considered very reliable, then records with influential errors can also be edited automatically. Limiting interactive editing to those records that likely contain influential errors which cannot be reliably resolved automatically is known as *selective editing*. This selection process is discussed in Chapter 4 of this report.

Selective editing makes use of expected values for the variables in a record to determine if these values deviate from one another. Strongly deviating values can be caused by an influential error. In determining these expected values, information is used from sources other than the actual file. Oftentimes, data from a previous period for the same statistic is used for this purpose. As such, it is possible to start the selection process for manual editing during the data collection period, as soon as the first records are received. Once all or most of the data has been received, then suspect values can also be detected by examining provisional estimates of totals and observations that exert a large influence on this. This form of selection is called macro editing (see below).

The automatic correction of random errors and other errors for which the cause cannot be established takes place in two steps. First, the best possible determination is made of what scores of variables in a record are incorrect. This is trivial if a value does not fall in the permissible range, such as a negative income or an improperly missing value. As such, the value is then certainly incorrect. In many cases, however, there are inconsistencies (violations of edit rules), for which is not clear which value or values are responsible. If, for example, an additive property rule (such as: the total staff expenditures are equal to the sum of the salaries, insurance premiums, training costs and other staff expenditures) is not satisfied, it is not clear what value or values in that sum are responsible for the violation of the rule. In the automatic detection of errors, the incorrect values are designated (the localisation of errors) according to the Fellegi-Holt paradigm, which states: designate as few as possible values as incorrect, but ensure that changing these values can result in a fully consistent record that satisfies all hard edit rules. The automatic localisation of errors based on the Fellegi-Holt paradigm is addressed in Chapter 5 of this report. Once the errors have been detected, they are replaced by better values by means of imputation. Automatic imputation takes place using models that can predict the incorrect or missing values. We will discuss this in detail in the theme *Imputation*.

An alternative method that localises automatic errors *and* imputes new values, the 'Nearest-neighbour Imputation Methodology', is discussed in Chapter 6 of this report.

Finally, in the last phase, provisional publication figures are calculated and analysed using historic data or external sources. This analysis is also called approval or macro editing, and will be discussed in Chapter 7 of this report. If the aggregate figures are implausible, the individual records are examined by, for example, further analysing outliers or influential records and correcting these as necessary. The errors detected at this phase may be errors that were not found in earlier phases of the detection and correction process or errors that were actually introduced by the process. In macro editing, the detection of errors begins at macro level, but the correction always takes place in the individual records, therefore at micro level. If the provisional figures are plausible, the detection and correction process is concluded.

The process in Figure 1 should be viewed as a prototype. In practice, not all of the steps will be undertaken for the different statistics. For example, there are few edit rules present for social statistics. As a result, the emphasis of the detection and correction process in that case will focus more on the correction for item non-response by means of imputation. For statistics based on registers, all the data (or a large amount of data) often becomes available at the same time. In that case, the macro editing can be started immediately. Also, when selecting records for manual editing, other criteria are often used than only whether a record contains influential errors. As such, important companies are frequently identified as crucial, and their data is always inspected manually. Examples of such companies could be those that are individually responsible for a significant portion of turnover in their sector. A decision can also be made to automatically edit very good imputable variables even if they potentially contain influential errors.

1.2 Scope and relationship with other themes

This theme addresses the detection and possible correction of potential errors in microdata. The intention of this part of the statistical process is to transform 'raw' microdata with errors and inconsistencies into corrected 'edited' microdata that is suitable for estimating publication figures and further analyses. The estimation itself, with its related problems such as determining raising weights, correcting for non-response by weighting and dealing with correct outliers, is described in other themes in the Methods Series.

In the theme *Data editing: Detection and Correction of Errors*, we discuss different methods developed to detect errors. This theme also addresses correction techniques that are used for the correction of errors in which the filled-in incorrect value contains information about the correct value, such as with systematic errors which can be explained. In addition, this theme addresses manual correction by experts (interactive editing). Correction for item non-response and incorrect values, for which the filled-in value does not contain any information about the correct value

and therefore has been interpreted as missing, often takes place by means of imputation. This subject will be discussed in the theme *Imputation*.

1.3 Place in the statistical process

As shown in Figure 1, the detection and correction process starts with raw data. This is the information as received from respondents and which is then checked, for written surveys, and stored in a standard format. In electronic questionnaires and CATI observations, certain checks can be conducted during the observation phase, which could lead to corrections by the respondent. However, this data is also considered as raw data for the detection and correction process, which begins after the observation phase.

As described in Section 1.1.2, most detection and correction procedures are performed per reporter, without knowledge of the answers from the other reporters being needed (however, this does not apply for the selection of influential errors at macro level). The detection and correction process can also be partially conducted during the data collection phase. This is important primarily for statistical processes in which a substantial amount of records are edited interactively. For these processes, from the perspective of the timeliness of the figures to be provided, it is important to start the editing process as early as possible. In statistical processes that involve little interactive editing, or which have a short data collection period, the detection and correction procedure can also begin after the data collection period.

Both the ‘input’ and the ‘output’ of the detection and correction process consist of a file with records per reporter. The detection and correction process transforms raw microdata with obvious errors, inconsistencies and missing values into edited microdata in which these problems have been resolved as far as possible. The edited file is used in the subsequent statistical process for aggregation purposes, for the estimation of totals and developments and for further analyses. The detection and correction process only makes changes to the microdata. Corrections of aggregate figures, such as are made for national accounts, are not part of detection and correction, but are part of a subsequent step in the statistical process.

1.4 Definitions

Concept	Description
automatic editing	A collective name for editing methods in which a computer program both detects and corrects the data
deductive correction	A collective name for editing methods in which the necessary corrections can clearly be derived from the uncorrected data
detection and correction	The detection and improvement of missing and incorrect values in a data file
editing	See ‘detection and correction’
edit rule	A restriction of the values in a data file; data that does not satisfy an edit rule will contain errors either with certainty (see ‘hard edit rule’), or with high probability (see ‘soft edit rule’)
hard edit rule	An edit rule that indicates with certainty that there is an error in the data
interactive editing	An editing method in which a computer program checks the data

	and a human editor corrects the data
influential errors	Errors that have a significant influence on the figures to be published
macro editing	A collective name for editing methods in which the detection of the data takes place on an aggregate level
manual editing	See ‘interactive editing’
micro editing	A collective name for editing methods in which detection and correction takes place at the level of the individual records
score function	An indicator of the influence that the interactive editing of a record is expected to have on the figure to be published; score functions are used to prioritise records for interactive editing (see ‘selective editing’)
selective editing	A collective name for methods to select records that contain possible influential errors for interactive editing; a score function is often used in this process
soft edit rule	An edit rule that indicates with high probability that there is an error in the data; data that does not satisfy a soft edit rule is suspect, but not necessarily incorrect

2. Methods for deductive correction

2.1 Short description

Data collected for compiling statistics frequently contains obvious systematic errors; in other words, errors that are made by multiple respondents in the same, identifiable way. Such a systematic error can often be detected automatically in a simple manner, in particular in comparison to the complex algorithms that are needed for the automatic localisation of non-systematic errors (see Chapters 5 and 6). Furthermore, after a systematic error has been detected, it becomes immediately clear which correction is necessary to restore it. For we know, or think we know with sufficient certainty, how the error came about.

A separate detection and correction rule is needed for each type of systematic error. The exact form of the correction method varies per type of error; there is no standard formula. This chapter therefore deviates slightly in terms of structure from the rest of the report. Most of the chapter will be used to discuss practical examples (Section 2.4) instead of general theory (Section 2.3).

The difficulty with using this method lies mainly in determining *which* systematic errors will be present in the data, before this data is actually collected. This can be studied based on data from the past. When certain edit rules are violated repeatedly in the same way, there is potentially a systematic error. Sometimes, such an investigation can bring systematic errors to light that have arisen due to a shortcoming in the questionnaire design or a bug in the processing procedure. In that case, the questionnaire and/or the procedure must be adapted. To limit the occurrence of discontinuities in a published time series, it can be desirable to ‘save up’ changes in the questionnaire until a planned redesign of the statistic, and to resolve the systematic error with a deductive correction method until that time.

2.2 Applicability

Deductive correction can be used on both quantitative and qualitative variables. Deductive methods are initially used to correct systematic errors. Such methods are generally not suitable to correct non-systematic (or random) errors. It is recommended that systematic errors are dealt with as far as possible at the start of the detection and correction process, before any other correction methods are used. In any case, it is known how these errors arose and how they can be reversed. The rest of the detection and correction process will proceed more efficiently after the systematic errors have been resolved deductively.

Errors for which the cause is known with sufficient certainty can be resolved deductively. In the case of incorrect assumptions about the error source, the method can lead to bias in the estimators. In practice, correction rules (see Section 2.3.1) can also be used on non-systematic errors for reasons of efficiency, if the introduced

bias is negligible. An example of this is the deductive resolution of rounding errors (cf. Scholtus, 2008a).

Systematic errors can often be identified by examining frequently occurring violations of edit rules. Deductive methods are therefore mainly effective for data for which many edit rules have been defined.

2.3 Detailed description

2.3.1 Correction rules

The simplest deductive correction methods can be represented as a single rule:

if (*condition*) **then** (*change*). (2.3.1)

Here, *condition* indicates a combination of values in a record that is not permissible. Subsequently *change* describes the correction that is made to resolve the inconsistency. Rules of this type are known as *correction rules*.

An example of a correction rule is:

if (*gender* = man and (*pregnant* = yes **or** *pregnant* = 'empty'))
then *pregnant* = no. (2.3.2)

This rule corrects records that do not satisfy the edit rule

if *gender* = man **then** *pregnant* = no.

Take note that the **if-then** construction is used here in two different ways. In an edit rule, the construction describes a condition that the data should satisfy; in a correction rule, it describes an action that results in changes in the data.

Another example of a correction rule is:

if (*age* < 18 and (*driving licence* = yes **or** *driving licence* = 'empty'))
then *driving licence* = no. (2.3.3)

This rule can be used to correct records that do not satisfy the edit rule

if *age* < 18 **then** *driving licence* = no.

Furthermore, more general deductive correction methods can, in principle, also be expressed as rules in the form (2.3.1). The **if** condition may then also contain information from other records or even from outside the data file to be corrected. The detection criterion can be rather complex; see the examples in Section 2.4.

2.3.2 Drawing up deductive corrections

A deductive correction method is intended to resolve an inconsistency that can only be resolved in a single way on logical and/or content-related grounds, under a certain assumption. If the assumption is correct, the deductive correction method always produces the actual values. The correction rule (2.3.2) operates, for example, under the assumption that the variable *gender* never contains errors. The same

applies for rule (2.3.3) and the variable *age*. These assumptions can for instance be satisfied if *gender* and *age* originate from a properly maintained population register.

Deductive correction methods are attractive because of their simplicity. However, they may only be used when no important nuances are lost with such a simple approach. If the data does not satisfy the assumptions made, deductive correction leads to biased estimators. For example: if *gender* or *age* has an incorrect value in some records, then, after applying the correction rules (2.3.2) and (2.3.3), we will underestimate the number of pregnancies or the number of driving licence holders respectively in the population.

Generally, a given inconsistency can be explained in many different ways. Even in the simple examples from Section 2.3.1, with only two variables, we can only correct the inconsistencies deductively by ruling out some explanations in advance. Deductive correction is generally only applicable if one of the explanations for the inconsistency is much more obvious than all the other possible explanations. To assess this, knowledge about the content of the data is often necessary.

An idea that is frequently used (sometimes subconsciously) when drawing up deductive correction methods is the following: if, for a given inconsistency, there is a correction that changes very little with respect to the current values, then it is highly probable that this will produce the actual values. Here, ‘changes very little’ can relate both to the number of changes and the nature of the changes. This is, in fact, a naïve version of the Fellegi-Holt paradigm. (See Chapter 5 for the real Fellegi-Holt paradigm.)

To illustrate, the first column of Table 1 shows a record that is inconsistent with respect to the edit rule

$$turnover - costs = profit. \tag{2.3.4}$$

The inconsistency can be resolved by adapting one of the three variables. The other columns of Table 1 show which possible changes this produces (the adapted value is shown in bold in each case). Intuitively, the solution in which *costs* is adapted is the most attractive, because changing the value 283 to 238 is less drastic than the other proposed corrections. Conversely, it is much more probable that the actual value of 238 was changed to 283 at some point during the collection and processing of the data, than the case that 398 was changed to 353 or 70 to 115. Therefore, we could draw up the following rule for deductive correction: if a record does not satisfy (2.3.4), but if it does when the digits in one of the amounts provided are reversed, then the record should be corrected in this way. When drawing up this correction method, we have used the ‘naïve Fellegi-Holt paradigm’ twice: first by not looking for solutions in which more than one variable is adapted, and then by choosing the least drastic of the remaining solutions.

This principle is addressed several more times in the practical examples in Section 2.4.

Table 1. Example of a record to be corrected deductively

	record	correction 1	correction 2	correction 3
turnover	353	**398**	353	353
costs	283	283	**238**	283
profit	115	115	115	**70**

2.3.3 Detecting unknown systematic errors

New systematic errors can be detected by analysing violations of edit rules. If an edit rule is violated frequently, this can be an indication of the presence of a systematic error in the relevant variables. A further analysis of the records that violate the edit rule, in which the questionnaire is also examined, can bring the cause of the error to light. Once the error has been identified, it is generally quite simple to draw up a deductive method to automatically detect and correct the error.

Detecting new systematic errors can only take place once sufficient data has been collected. The results are therefore usually too late for the current observation period. If the analysis produces new deductive correction methods, then these can be built in to the correction process for the data in the next observation period.

As far as systematic errors are concerned, prevention is better than cure. Sometimes it is possible to improve the design of the questionnaire so that far fewer respondents make a certain type of error. If many respondents make errors in the same way, this can be an indication that the questions are not presented clearly enough. In some cases, it is also possible to adapt the processing procedure to ensure that a certain processing error no longer arises. In principle, this approach should be preferred to that of making deductive corrections afterwards. However, because there are practical objections to the constant adaptation of the questionnaire, it is sufficient initially to build in a deductive correction method, and to include the knowledge gained in a later redesign of the questionnaire.

To illustrate this, we detect a new systematic error in the data of the Structural Business Statistics (SBS) for Wholesale 2001. One of the many edit rules is as follows:

LOONSOM110000 + LOONSOM121100 + LOONSOM121200 +
LOONSOM122000 = LOONSOM100000.

Here, LOONSOM100000 (‘*loonsom*’ means ‘payroll total’) represents the total labour costs. The other four variables are the sub-items of this total.

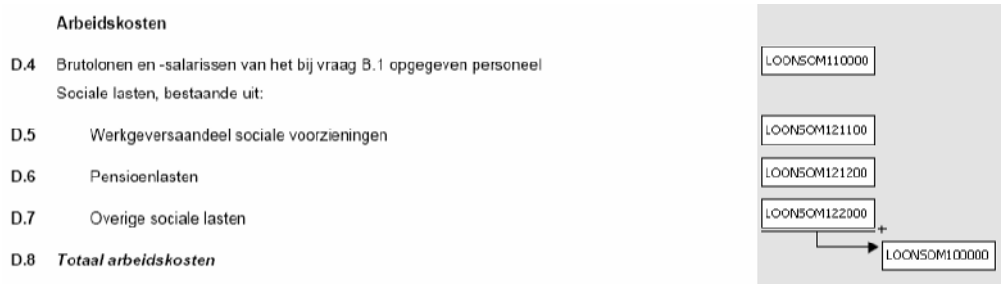
Table 2 shows several records that violate the edit rule.

Table 2. Examples of inconsistent records in the SBS for Wholesale 2001

	record 1	record 2	record 3	record 4
LOONSOM110000	1 100	364	1 135	901
LOONSOM121100	88	46	196	134
LOONSOM121200	40	34	68	0
LOONSOM122000	42	0	42	0
LOONSOM100000	170	80	306	134

It is striking that, in these records, the items LOONSOM121100, LOONSOM121200 and LOONSOM122000 add up to the total LOONSOM100000. This means that it seems that these reporters have ignored the first sub-item LOONSOM110000 in the calculation of LOONSOM100000. A closer look at the questionnaire (Figure 2) makes it clear why this happened: there is a gap between the answer box for LOONSOM110000 and the other boxes. As a result, from how the question is phrased, it cannot be clearly determined whether LOONSOM110000 is part of the sum or separate from the rest. Most respondents understand from the context what the intention is, but in several dozen records, we see the error from Table 2.

Figure 2. Part of the questionnaire SBS Wholesale (to 2005)



We can draw up a correction method that resolves this error deductively. A more structural solution consists of removing the cause of the error by adapting the questionnaire. This has already been done: the questionnaire from Figure 2 was replaced by the SBS 2006. On the new questionnaire, the answer boxes are spaced evenly.

2.4 Examples

This section addresses several practical examples of methods already used or at least developed for deductive correction. We will first discuss examples from the statistic Building Objects in Preparation (Section 2.4.1) and Short Term Statistics (Section 2.4.2). The other examples originate from the SBS. We will first discuss the deductive correction methods in the current processing procedure (Section 2.4.3) and then three recently developed methods (Sections 2.4.4 to 2.4.6).

2.4.1 Correction rules for the statistic Building Objects in Preparation

The quarterly statistic Building Objects in Preparation (BOP) follows the development of the total construction value of new contracts at architectural firms in the Netherlands. In 2007, a new detection and correction process was designed for this statistic: see Van der Loo and Pannekoek (2007), from which this example is taken.

When filling in the BOP questionnaire, the reporter must answer several questions about each building object separately. The reporter must tick a box indicating whether the building objects concerns a residence (*r*), a combined-purpose building[1] (*c*) or neither of these (*o* for other). Another question concerns *n*, the total number of dwellings in the building. For a combined-purpose building, the percentage of floor area intended for residential use (*p*) is also requested.

The statement contains an error if zero, two, or three of the boxes for *r*, *c* and *o* have been ticked. In that case, the type of building object has not been clearly specified. In certain situations, this error can be deductively corrected based on *n* and *p*.

If the value indicated for *n* is greater than zero and moreover if *p* is equal to 100% or is not filled in, it is obvious that the building object is a residence. If *n* is larger than zero and furthermore if *p* is not equal to 0 or 100%, it is obvious that the building object is a combined-purpose building. And, finally, if neither *n* nor *p* has been filled in, or if they have been given the value of 0, then it is highly probable that the building object falls in the category 'other'. These interpretations follow from the assumption that the statement must be rendered correct by changing as few values as possible.

We write $r = \mathsf{T}$ if the box for residence has been ticked, and otherwise $r = \mathsf{F}$, and we do the same for *c* and *o*. The correction rule is now as follows:

if $(r,c,o) \in \{ (\mathsf{T},\mathsf{T},\mathsf{T}) , (\mathsf{T},\mathsf{T},\mathsf{F}) , (\mathsf{T},\mathsf{F},\mathsf{T}) , (\mathsf{F},\mathsf{T},\mathsf{T}) , (\mathsf{F},\mathsf{F},\mathsf{F}) \}$
then
if (p = 'empty' **or** $p = 100\%$) and $n > 0$
then $(r,c,o) = (\mathsf{T},\mathsf{F},\mathsf{F})$
if $0\% < p < 100\%$ and $n > 0$
then $(r,c,o) = (\mathsf{F},\mathsf{T},\mathsf{F})$
if (p = 'empty' **or** $p = 0\%$) and (n = 'empty' **or** $n = 0$)
then $(r,c,o) = (\mathsf{F},\mathsf{F},\mathsf{T})$.

This is a small part of the detection and correction process for the statistic BOP.

In the implementation of the detection and correction process for BOP, the derivation of the correction always takes place separately from the actual application of the correction. Initially, in the above example, only an indicator is created that specifies for each record whether a deductive correction is applicable, and if so, which one. Only in the next step are the values of *r*, *c* and *o* changed in the record.

[1] A combined-purpose building is used for other purposes in addition to residential purposes.

As such, the detection and correction process is transparent, so that it is clearly visible afterwards exactly what changes have been made to each record.

2.4.2 Correction of thousand-errors in Short Term Statistics

Company surveys usually contain instructions for the reporter that all financial amounts must be rounded to thousands of euros. Some respondents ignore these instructions and give values that are a factor 1000 larger than they actually mean. It is clear that, if these 'thousand-errors' are not corrected, the resulting estimates for the figures to be published will be too high.

We refer to a *uniform thousand-error* if all the financial amounts in a record are too large by a factor of 1000. It is known that, mainly in longer questionnaires, there are also records in which a non-uniform (or partial) thousand-error occurs. A non-uniform thousand-error can arise if several people each fill out part of the questionnaire.

Thousand-errors are detected by comparing one or more amounts provided with reference values. The reference data used and the way in which the comparison takes place varies per statistic and per statistics bureau. Examples of reference data are: a statement from the same respondent from an earlier period, the median of a number of similar respondents in an earlier period and the register data about the respondent. It is important that this reference data has been previously checked for errors.

For Short Term Statistics (STS), thousand-errors are detected as follows (Ter Haar, 2002). The total turnover indicated by the respondent is compared to the turnover from the most recent period for which a statement from the respondent is available, up to a maximum of six previous periods. The stated turnover for this earlier period must also not be equal to zero. There is a thousand-error if the following applies (where abs(a) is the absolute value of a):

$$\text{abs}(turnover_t) > 300 \times \text{abs}(turnover_{t-i}) > 0, \qquad \text{for some } i \in \{1,\ldots,6\}.$$

If no data from the respondent from an earlier period is available, then the median of the turnover from the previous period in the stratum of the respondent is examined. There is a thousand-error if the following applies:

$$\text{abs}(turnover_t) > 100 \times \text{stratum median}(turnover_{t-1}).$$

If a thousand-error is detected in this way, it is resolved by dividing the total turnover and all the sub-items by 1000.

Table 3 shows an example of a record with a thousand-error that was found in this way.

Table 3. Example of a uniform thousand-error

	reference data	data before correction	data after correction
first sub-item turnover	3 331	3 148 249	3 148
second sub-item turnover	709	936 142	936
total turnover	4 040	4 084 391	4 084

2.4.3 Correction of systematic errors at the SBS

The SBS questionnaire contains a large number of financial variables that must satisfy a variety of edit rules, such as sub-items that add up to a total and ratios that fall within certain bounds. This set of edit rules is a rich source for detecting systematic errors in the data. To date (SBS 2007), eight correction methods for systematic errors have been implemented, and we will discuss three here. Starting in Section 2.4.4, three methods will be examined that will probably be used in the future.

The most important systematic error that is automatically corrected by the SBS is the uniform thousand-error. The approach is similar to that of the STS (see Section 2.4.2). Instead of directly comparing the turnover with an earlier period, the ratio between the stated turnover and the stated number of people employed is examined. A thousand-error is detected when this ratio strongly deviates from the stratum median in the previous period, i.e. if

$$turnover_t / pe_t > 100 \times \text{stratum median}(turnover_{t-1} / pe_{t-1}), \qquad (2.4.1)$$

where *pe* is the number of people employed.[2] In addition, the VAT register data and the STS data are used as a reference. For the respondents for which a positive VAT or STS annual turnover *turnover_external* is known, it is determined whether the following applies:

$$turnover_t > 100 \times turnover\_external_t.$$

If yes, then a thousand-error is detected. In both cases, all the financial amounts stated are divided by 1000.

[2] SBS documentation has revealed that, instead of (2.4.1), the following formula was used:

$$turnover_t / pe_t > 100 \times \text{stratum median}(turnover_{t-1}) / \text{stratum median}(pe_{t-1}).$$

In general, however, the median of the ratios is not equal to the ratio of the medians. A simple example:

$$\text{median}(\{ 1 , 10^6 , 10^6 \}) / \text{median}(\{ 1 , 1 , 10^6 \}) = 10^6 / 1 = 10^6,$$

while

$$\text{median}(\{ 1 / 1 , 10^6 / 1 , 10^6 / 10^6 \}) = \text{median}(\{ 1 , 10^6 , 1 \}) = 1.$$

A second systematic error at the SBS concerns incorrectly placed minus signs. If a value must be subtracted, some respondents indicate this by placing a minus sign before the stated amount. This takes place despite the fact that there is already a printed minus sign on the survey form. After keying in, the variable incorrectly has a negative value. This error is resolved by using the absolute value of the number filled in.

Furthermore, the SBS examines records in which sub-items are filled in, while the accompanying total is empty. This error is corrected by calculating the total based on the edit rule that says that the sub-items must add up to the total. This correction assumes that any empty sub-items have a value of zero. Pannekoek and Tempelman (2005) demonstrate that this assumption is not always satisfied.

2.4.4 Correction of sign errors and interchanged returns and costs

The profit and loss account is a part of the SBS questionnaire where many errors are made. The account is composed of balances that must add up to an end balance. In addition, some balances are divided into returns and costs. Stated generally, this means that the following edit rules apply:

$$\begin{cases} x_0 = x_{0,r} - x_{0,c} \\ \quad \vdots \\ x_m = x_{m,r} - x_{m,c} \\ x_n = x_0 + x_1 + \cdots + x_{n-1} \end{cases} \tag{2.4.2}$$

In this example, $x_0, x_1, \ldots, x_{n-1}$ represent the balances, x_n the end balance, and $x_{k,r}$ and $x_{k,c}$ the returns and costs that are associated with balance x_k. To keep the notation simple, we assume that only $x_0, \ldots, x_m$ are divided, for some $m \in \{0,1,\ldots,n-1\}$. The bottom rule from (2.4.2) is called the *external sum rule*, while the other rules are known as *internal sum rules*.

Table 4 shows the structure of the profit and loss account from the questionnaire that was used to 2005 inclusive by the SBS. The edit rules are indicated by (2.4.2) with $n = 4$ and $m = n - 1 = 3$. Table 4 also contains three examples of records that are inconsistent with respect to (2.4.2).

In example (a), two edit rules are violated: the external sum rule and the internal sum rule of the financial result. Remarkably enough, we can cancel out both violations by solely changing the value of x_1 from 10 to –10 (see Table 5). The obvious choice is to correct the record in this way, because any other solution would require changing more than one value.

Table 4. Examples of sign errors and interchanged returns and costs

Variable	Name	(a)	(b)	(c)
$x_{0,r}$	*total operating income*	2 100	5 100	3 250
$x_{0,c}$	*total operating costs*	1 950	4 650	3 550
x_0	*operating result*	150	450	300
$x_{1,r}$	*financial returns*	0	0	110
$x_{1,c}$	*financial costs*	10	130	10
x_1	*financial result*	10	130	100
$x_{2,r}$	*withdrawals and release from provisions*	20	20	50
$x_{2,c}$	*additions to provisions*	5	0	90
x_2	*balance of provisions*	15	20	40
$x_{3,r}$	*extraordinary returns*	50	15	30
$x_{3,c}$	*extraordinary costs*	10	25	10
x_3	*extraordinary result*	40	10	20
x_4	*pre-tax result (end balance)*	195	610	–140

*Table 5. Corrected version of Table 4. Adapted values are shown in **bold**.*

Variable	Name	(a)	(b)	(c)
$x_{0,r}$	*total operating income*	2 100	5 100	3 250
$x_{0,c}$	*total operating costs*	1 950	4 650	3 550
x_0	*operating result*	150	450	**–300**
$x_{1,r}$	*financial returns*	0	**130**	110
$x_{1,c}$	*financial costs*	10	**0**	10
x_1	*financial result*	**–10**	130	100
$x_{2,r}$	*withdrawals and release from provisions*	20	20	**90**
$x_{2,c}$	*additions to provisions*	5	0	**50**
x_2	*balance of provisions*	15	20	40
$x_{3,r}$	*extraordinary returns*	50	**25**	30
$x_{3,c}$	*extraordinary costs*	10	**15**	10
x_3	*extraordinary result*	40	10	20
x_4	*pre-tax result (end balance)*	195	610	–140

In example (b), two internal sum rules are violated. The obvious way to make this record consistent is: interchange the values of $x_{1,r}$ and $x_{1,c}$ and interchange the values of $x_{3,r}$ and $x_{3,c}$ (see Table 5). This correction uses amounts filled in by the respondent and therefore is preferred to a solution in which synthetic values are imputed.

The types of inconsistencies in example (a) and (b) are called *sign errors* and *interchanged returns and costs* respectively. For the sake of brevity, we also use

'sign error' as the umbrella term. These errors are related and therefore must be detected simultaneously.

There is a sign error if the following two conditions have been satisfied:

- The record does not satisfy (2.4.2).
- The record can be adapted by only changing the signs of balances and interchanging returns and costs, so that it does satisfy (2.4.2).

In this process, the total operating income ($x_{0,r}$) and total operating costs ($x_{0,c}$) must not be interchanged, because these are also associated with items outside the profit and loss account by means of other edit rules than (2.4.2). Furthermore, because of the structure of the questionnaire, it is highly improbable that the respondent would confuse these two answers.

A mathematical formulation of the above conditions is that an inconsistent record contains a sign error if the following system of equations has a solution $(s_0,\dots s_n;t_1,\dots,t_m)\in\{-1,1\}$:

$$\begin{cases} x_0 s_0 = x_{0,r} - x_{0,c} \\ x_1 s_1 = (x_{1,r} - x_{1,c})t_1 \\ \quad\vdots \\ x_m s_m = (x_{m,r} - x_{m,c})t_m \\ x_n s_n = x_0 s_0 + x_1 s_1 + \cdots + x_{n-1} s_{n-1} \end{cases} \tag{2.4.3}$$

If such a solution is found, it is also immediately clear how the sign error can be corrected. For each $s_j = -1$, the sign of the accompanying x_j must be changed, and for each $t_k = -1$, $x_{k,r}$ and $x_{k,c}$ must be interchanged. It is not difficult to see that the resulting record satisfies (2.4.2). We point out that a variable t_0 is missing in (2.4.3) because $x_{0,r}$ and $x_{0,c}$ are not eligible to be interchanged. Apart from the motivation presented above, there is also a technical reason for this: the first equation in (2.4.3) now uniquely fixes the value of s_0. By fixing one of the variables, we prevent that we can reconstruct a solution to (2.4.3) to an alternative solution by multiplying all variables by –1.

Applied to example (c) from Table 4, (2.4.3) becomes:

$$\begin{cases} 300 s_0 = -300 \\ 100 s_1 = 100 t_1 \\ 40 s_2 = -40 t_2 \\ 20 s_3 = 20 t_3 \\ -140 s_4 = 300 s_0 + 100 s_1 + 40 s_2 + 20 s_3 \end{cases}$$

This system has the following solution:

$$(s_0 = -1, s_1 = 1, s_2 = 1, s_3 = 1, s_4 = 1; t_1 = 1, t_2 = -1, t_3 = 1).$$

Therefore, example (c) contains a sign error. To correct this, we must change the value of x_0 from 300 to –300, and interchange the values of $x_{2,r}$ and $x_{2,c}$. Table 5 shows that these changes do indeed produce a consistent record.

In summary, the method to detect and correct sign errors and interchanged returns and costs in the SBS profit and loss account is as follows:

1. Given a record that does not satisfy (2.4.2), determine system (2.4.3).
2. Find the[3] solution $(s_0, \ldots s_n; t_1, \ldots, t_m) \in \{-1,1\}$ of (2.4.3), if it exists. Stop if (2.4.3) does not have a solution, and otherwise continue with step 3.
3. For $j = 0, \ldots, n$ and for $k = 1, \ldots, m$: change the sign of x_j if $s_j = -1$, and interchange the values of $x_{k,r}$ and $x_{k,c}$ if $t_k = -1$.

The only non-trivial step in this scheme is step 2, the resolution of system (2.4.3). Given that n and m are small, the solution can be found in principle by systematically trying all $2^{n+m+1} - 1$ combinations of $s_0, \ldots, s_n$ and $t_1, \ldots, t_m$. In Scholtus (2007), the solution of (2.4.3) is rewritten as a binary linear programming problem that can be resolved with standard software.

2.4.5 Correction of cumulation errors

Another error that regularly occurs in the SBS profit and loss account is the so-called *cumulation error*. Table 6 shows three examples of records with a cumulation error. The error occurs because the respondent fills in the profit and loss account 'cumulatively'. In example (a) and (b), this occurs consistently, but this is not the case in example (c). Furthermore, financial returns and costs are also interchanged in example (c).

Say that a given record does not satisfy the external sum rule or the k^{th} internal sum rule, for some $k \in \{1, \ldots, n-1\}$, but that the following does apply:

$$x_k = x_{k-1} + x_{k,r} - x_{k,c}. \tag{2.4.4}$$

In that case, there is a cumulation error, which can be corrected by replacing the values of x_k, $x_{k,r}$ and $x_{k,c}$ by

$$x'_k = x_k - x_{k-1}, \qquad x'_{k,r} = x_{k,r}, \qquad x'_{k,c} = x_{k,c}.$$

[3] Appendix A of Scholtus (2008a) proves that (2.4.3) has no more than a single solution under very mild conditions.

Table 6. Examples of cumulation errors

Variable	Name	(a)	(b)	(c)
$x_{0,r}$	*total operating income*	6 700	8 300	6 900
$x_{0,c}$	*total operating costs*	5 650	5 400	6 150
x_0	*operating result*	1 050	2 900	750
$x_{1,r}$	*financial returns*	0	0	0
$x_{1,c}$	*financial costs*	0	150	40
x_1	*financial result*	1 050	2 750	790
$x_{2,r}$	*withdrawals and release from provisions*	0	0	0
$x_{2,c}$	*additions to provisions*	0	30	0
x_2	*balance of provisions*	1 050	2 720	0
$x_{3,r}$	*extraordinary returns*	0	0	0
$x_{3,c}$	*extraordinary costs*	0	110	0
x_3	*extraordinary result*	1 050	2 610	0
x_4	*pre-tax result (end balance)*	1 050	2 610	790

From (2.4.4), it follows immediately that $x'_k = x'_{k,r} - x'_{k,c}$. By performing this step successively for each $k \in \{1,\ldots,n-1\}$, example (a) and (b) can be made fully consistent. Here, incidentally, the *original* value of x_{k-1} must be substituted in (2.4.4) for all *k*, and not x'_{k-1}.

To also take account of possible sign errors, instead of (2.4.4), it must be examined whether $(\lambda,\mu) \in \{-1,1\}$ exist, such that the following applies:

$$\lambda x_k = x_{k-1} + \mu(x_{k,r} - x_{k,c}). \tag{2.4.5}$$

If so, then the record contains a cumulation error. If $\lambda = -1$, then x_k also has an incorrect sign, and if $\mu = -1$, then $x_{k,r}$ and $x_{k,c}$ have been interchanged. The cumulation error *and* the sign error are corrected by replacing x_k, $x_{k,r}$ and $x_{k,c}$ by:

$$\begin{cases} x'_k = \lambda x_k - x_{k-1} \\ x'_{k,r} = \dfrac{1+\mu}{2} x_{k,r} + \dfrac{1-\mu}{2} x_{k,c} \\ x'_{k,c} = \dfrac{1-\mu}{2} x_{k,r} + \dfrac{1+\mu}{2} x_{k,c} \end{cases} \tag{2.4.6}$$

Note that $x'_{k,r} = x_{k,r}$ if $\mu = 1$ and $x'_{k,r} = x_{k,c}$ if $\mu = -1$, and something similar for $x'_{k,c}$. From (2.4.5), it follows that $x'_k = x'_{k,r} - x'_{k,c}$.

For example: in example (c), we see that

$$1 \cdot x_1 = 790 = 750 - (-40) = x_0 + (-1) \cdot (x_{1,r} - x_{1,c}),$$

in other words, (2.4.5) applies for $(\lambda = 1, \mu = -1)$. According to (2.4.6), the error can be corrected by selecting: $x_1' = x_1 - x_0 = 40$, $x_{1,r}' = x_{1,c} = 40$ and $x_{1,c}' = x_{1,r} = 0$. Now $x_1' = x_{1,r}' - x_{1,c}'$ does indeed apply. Because there are no other errors in this record, this correction immediately satisfies the external sum rule.

A slightly more detailed elaboration of this method can be found in Scholtus (2008a).

2.4.6 Correction of simple typing errors

In Section 2.3.2, we saw an example in which an inconsistency was resolved deductively by assuming that the respondent had accidentally interchanged two digits (by writing '283' instead of '238'). Interchanging two subsequent digits is an example of a simple typing error. Other examples are:

- Adding a digit (for example: '46297' instead of '4627');
- Forgetting a digit (for example: '427' instead of '4627');
- Replacing a digit (for example: '4687' instead of '4627').

Simple typing errors are easy to make and therefore occur frequently in practice. A review of data from the SBS Wholesale 2007 revealed, for example, that nearly 10% of all inconsistencies in linear equations could be explained by one of the four errors mentioned above (Scholtus, 2009).

In the event that the data must satisfy a single linear equation, simple typing errors can easily be detected, such as in the example from Section 2.3.2. Van de Pol et al. (1997) address this situation in detail. In the SBS edit rules, however, there is a system of linear equations that are connected with one another. In addition to edit rule (2.3.4), *turnover* and *costs*, for example, must be equal to the sum of several sub-items. A deductive method to correct simple typing errors in this more complex situation is described by Scholtus (2009). We will discuss this method here using only an example.

Say that a record consists of eleven variables that must satisfy five edit rules:

$$\begin{cases} x_1 + x_2 = x_3 \\ x_2 = x_4 \\ x_5 + x_6 + x_7 = x_8 \\ x_3 + x_8 = x_9 \\ x_9 - x_{10} = x_{11} \end{cases}$$

The following record violates the second, fourth and fifth edit rules:

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}
1452	116	1568	161	323	76	12	411	19979	1842	137

To see if one or more inconsistencies can be explained as simple typing errors, we first determine which variables only occur in violated linear equations. After all these are the only variables that we can change deductively without introducing new

inconsistencies. In this example these are the variables that only occur in the second, fourth and fifth edit rules, and these are x_4, x_9, x_{10} and x_{11}.

For each stated variable, we go through the linear equations in which it occurs. For each edit rule, we determine which value the variable in question should be given to eliminate the inconsistency. Please note: such a value always exists if the edit rule is in the form of a linear equation. Next, we compare the new value with the filled-in value. If the change can be explained as simple typing error, then we keep the proposed change, otherwise we do not. After going through all the edit rules, we look at how often each proposed value has been stated.

In this example, x_4 only occurs in the second edit rule. To satisfy this rule, we would have to fill in the value $\tilde{x}_4 = 116$. The current value is 161, and the new value can be explained as a simple typing error: two successive digits have been interchanged. This means that it is plausible that the actual value of 116 has been changed to the observed value of 161 due to a typing error.

Variable x_9 occurs in both the fourth and the fifth edit rules. Both rules can be satisfied by filling in the value of $\tilde{x}_9 = 1979$. This value can also be explained by a simple typing error: the actual value of 1979 was likely changed to the observed value of 19979 because an additional digit was accidentally added.

For x_{10}, we see $\tilde{x}_{10} = 19842$. From this value, the observed value of 1842 can be found by leaving out a digit. This could also be a simple typing error.

Finally, the necessary value of x_{11}, $\tilde{x}_{11} = 18137$, cannot be explained by one of the abovementioned simple typing errors. This means that we are not addressing x_{11} further here.

Next, a choice must be made from the possible typing errors found. It is clear that, for each variable, no more than one new value can be selected. Furthermore, it is not useful to change two variables that are present in the same linear equation: on balance, the edit rule continues to be violated. Given these two limitations, we choose the combination of proposed changes that leads to a maximum number of resolved inconsistencies.

In the example, we have found no more than one new value for each variable, therefore the first limitation does not play a role. Further consideration of the edit rules demonstrates that x_4 does not occur in the same rule as x_9 and x_{10}, but that x_9 and x_{10} do occur together in an equation. There are therefore two possible choices: either to change x_4 and x_9, or x_4 and x_{10}. The number of resolved inconsistencies in these choices is three and two respectively. We therefore choose the first combination of deductive corrections. The resulting record is:

x_1	x_2	x_3	$\tilde{x}_4$	x_5	x_6	x_7	x_8	$\tilde{x}_9$	x_{10}	x_{11}
1452	116	1568	116	323	76	12	411	1979	1842	137

This record satisfies all the edit rules.

Table of Contents

A detailed description of this algorithm for the deductive correction of simple typing errors can be found in Scholtus (2009).

2.5 Quality indicators

As previously stated, people always make assumptions about the data when drawing up a deductive correction method. If these assumptions are valid, the method produces the best possible corrections. However, if the assumptions are unrealistic, the method can introduce bias. It is therefore important to investigate whether the data satisfies the assumptions made.

An indicator of the usefulness of a deductive correction method is the number of errors that it resolves in a realistic data file. Another aspect concerns the gain in efficiency that is achieved because a number of records – after the implementation of the deductive method – need a lesser amount of detection and correction, or a less intensive form. An example of this can be found in the detection and correction process of the SBS, where there is a choice between manual (‘expensive’) and automatic editing (‘inexpensive’). The deductive corrections described in Section 2.4.3 create a situation where more records are suitable for the automated variant.

3. Interactive editing

3.1 Short description

In interactive or manual editing, an editor makes corrections to a record, using a program that detects errors and shows which variables are involved in a violated edit rule. It is then the editor's decision to select which variable should be corrected and what the correct value of this variable could be. At Statistics Netherlands and many other statistical bureaus, BLAISE is used for this purpose. For telephone or personal interviews, the editing can be started during the data entry. Drawing up editing instructions beforehand is recommended to guarantee the quality of the editing.

3.2 Applicability

To clarify the objective of interactive editing, a distinction is initially made among several types of values.

The *true value* is obtained through an ideal measurement and processing procedure, i.e. the value that we obtain if the reporter provides figures according to the right definition and using the right accounting records, and that are not changed during the processing procedure. However, we do not know the true value, and therefore we also do not know if we have observed it.

What we can attain is a *correct value*, namely if a sector expert considers this correct based on the available information, or on the information available at Statistics Netherlands about values of related variables and records and an extensive set of editing rules. Whether this 'correct' value approximates the true value depends on the available information, the skill of the editor, the quality of the editing instructions and the extent to which these instructions are followed.

In many cases, however, we cannot obtain anything better than an *acceptable value*, which means that it satisfies the hard edit rules. If an editor only resolves hard errors, then this produces an acceptable value. This also applies if a record is edited automatically. In that case, only hard errors are resolved; see Chapter 5.

The objective of interactive editing is to make values in a record correct. In the event of severe time pressure, a situation can arise where a limited number of variables are checked exhaustively and the other variables are made no more than acceptable during the micro editing phase. This choice must be made by the project manager, not the editor. Influential errors that remain in the micro editing must still be edited interactively during the macro editing.

Interactive editing is particularly effective if the data can only be partly edited automatically or if the quality of interactively edited data is significantly better. Another advantage is that, during interactive editing, you can search for information on, for example, the internet or in a completed written questionnaire. Reporters can be telephoned in case of primary observation. This is only recommended if it is

crucial for insight into the statistical process or the quality of a publication figure. Interactive editing also offers the opportunity to recognise error patterns that occur regularly. It can then be examined whether these error patterns can be automatically edited (in advance) in the subsequent process.

An important condition is that a set of edit rules is available with which mutual relationships and the range of variables (or ratios thereof) can be checked. In addition, a program should be available that can go through the edit rules and make a distinction between hard and soft errors. This program must also be able to show reference values for a record; see Section 3.3.

3.3 Detailed description

3.3.1 Introduction

When interviewing people, a form can be immediately edited interactively if a computer-assisted interviewing (CAI) system, such as BLAISE, is used. Inconsistencies can be detected by the CAI system and corrected by the interviewer in consultation with the interviewee. For companies, this is only possible if they are visited by the field staff. If a company submits a form, it is advisable to only interactively edit the form if it contains potential influential values. This can be determined using score functions; see Chapter 3. The editor can examine these scores to determine which variables contain potential influential errors.

If a survey completed by a reporter is received by Statistics Netherlands, an automatic correction round takes place first, during which obvious errors are corrected. In the interactive correction of *pre-edited* data, the remaining incorrect values are improved by contacting the reporter or by making use of expert knowledge in combination with reference data, such as other data from the same reporter (from a previous period, another survey or accounting records), the reporter's original statement or representative values of trends for similar reporters.

In the interactive correction of a survey with related variables, changing a value of a variable can result in the violation of other edit rules. In this case, other variable will also have to be corrected. In any case, an editor will have to ensure that the data satisfies all the hard edit rules. The editor must determine which variable in a violated edit rule must be corrected, and what the correct value is.

In the interactive correction of a short-term statistic, it can be examined, for example, whether an influential suspect value fits in the seasonal pattern for similar units. For economic statistics, account can be taken of the general picture of the economic development in recent periods.

3.3.2 Drawing up editing instructions

It is not sufficient to have a programme that shows the variables of the records to be checked, indicates why a record has been selected, which edit rules have been violated, and makes visible the related variables from other sources and earlier

periods or process steps. It is important to draw up editing instructions to prevent an editor from using an incorrect editing strategy.

Editing instructions must contain at least the following components:

- An explanation of the observation and processing procedure that has taken place.
- Instructions about the order in which the selected records should be dealt with. If the interactive editing is part of macro editing, then analysis instructions are also necessary. This indicates how records (and extra records) can be selected.
- For selective editing, an explanation is needed about the selection criterion and how this can be used to detect errors in a record.
- An overview of the type of errors that can occur in the data, such as NACE (Standard Industrial Classification) errors, size class errors, measurement errors and processing errors.
- Tips about detecting a certain error. In statistics about multiple related variables, you can search for scatter plots and complex units. This is also possible for other statistics if connections can be made with related variables from other sources. For short-term statistics, the seasonal pattern of a record can be compared with the seasonal pattern for the sector.
- Suggestions about additional information that can be looked up; for example, using a register of statistical units, sector organisations, the internet or Cdfoon. Googling a company name helps, for example, to determine whether there is a NACE error.
- For each type of error, there should be an indication of how the error can be corrected. A correction rule may be specified for systematic errors.
- Instructions about recording the editing actions taken; for example, using a comments field in the editing tool. If a NACE error or size class error has been observed, then it must be clear whether this must be communicated to the people that manage the population frame.

3.4 Quality indicators

To determine whether interactive editing will improve the microdata and publication figures, a number of aspects can be examined:

1. Percentage and number of records that do not satisfy an edit rule before interactive editing;
2. Percentage and number of records that do not satisfy an edit rule after interactive editing;
3. Publication figure calculated based on pre-edited data for interactively edited records and acceptable data for automatically edited records;
4. Publication figure calculated based on interactively edited data.

The difference between indicators 1 and 2 provides an understanding of the extent to which violations of edit rules are resolved by interactive correction. The difference between indicators 3 and 4 indicates the effect of interactive editing on the publication figure.

4. Selective editing

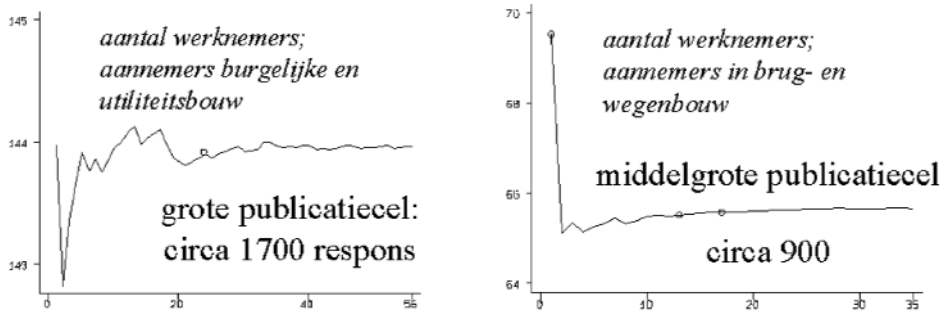
4.1 Short description

Manual or interactive editing is one of the most time-consuming and expensive parts of the statistical processing procedure for business statistics. In the past, all records were frequently manually edited, which led to high costs and long turnaround times. This was a stimulus for research into possibilities to limit the manual work. In the last two decades, this research has revealed that it is neither necessary nor desirable to edit *all* records manually. For many records, manual corrections have a negligible influence on the ultimate publication figures, such as estimates of population/sub-population totals or developments. Manual editing can therefore be limited to those records for which corrections do actually affect the publication figures; this focused restriction is known as selective editing. The intention of selective editing is to limit manual editing in order to reduce costs, decrease the turnaround time, and reduce the response burden with only a minimum loss of quality in the publication figures.

In selective editing, each record is assigned a score that indicates the extent of the expected influence on the publication figures if the record were manually edited. The records with high scores (significant influence) have the highest priority for manual editing. For low scores, under a certain limit value, manual editing is no longer necessary. Methods that can be used to determine the scores and methods to establish the limit value are part of the selective editing methodology that is discussed in this chapter.

The figures below illustrate the declining influence of successively less important corrections on the estimates of totals (taken from Hoogland et al., 2002). The estimated totals are shown as a function of the number of edited records, in which the records were edited in the order of decreasing influence on this estimate. The left diagram in Figure 3 concerns the estimate of the number of employees in the building industry in a subdomain with 1700 respondents. This figure shows that the correction of more than 40 records does not have any additional impact on the estimate of the population total. The right diagram in Figure 3 shows that the estimate of the number of employees in civil engineering in a subdomain with 900 respondents hardly changes once the 20 records with the largest errors have been corrected. These figures present examples for which manual editing can be limited to a small part of the records. The extent of manual editing is unlikely to remain as strongly limited in all situations as in these examples. However, from the perspective of efficiency, it is virtually always useful to utilise manual editing selectively. Methods to determine which records are suitable for manual editing and which records do not need this will be discussed in this chapter.

Figure 3. Estimated number of employees as a function of the number of edited records, for which editing was performed in the order of impact on the total. The first display is based on a large publication cell (about 1700 respondents), the second display is based on a medium-sized publication cell (about 900 respondents).



4.2 Applicability

Selective editing is used almost exclusively for business statistics and numerical variables. The impact of the editing on publication figures varies significantly between companies, simply because they vary – often significantly – in size, and therefore have a very different share in the estimate of the total. This is less true for social statistics. Each individual has approximately the same importance for the estimate of a total, and this importance is expressed in the raising weights that vary minimally between respondents. Still, individual records with strongly deviating values, such as extremely high incomes, can be detected and manually checked.

While it is worthwhile to perform selective editing for virtually all economic statistics, this does not always have to take place at micro level. An alternative is macro editing (see Chapter 7). An advantage of selective editing at micro level instead of macro editing is that the editing can be started during the data collection period. An advantage of editing at macro level is that the major errors can be better detected once all the data has been received.

In the processing procedure, the records are selected for manual editing after the systematic errors have been corrected (see Chapter 2). This is an automatic correction round in which major errors (such as thousand-errors) are often corrected. If such errors are not corrected first, they will be recognised during selective editing as influential errors and the records involved will be routed to the editors. Clearly, it is inefficient to burden the editors with these errors that can be resolved automatically.

4.3 Detailed description

4.3.1 *Introduction*

The most important tool in the selective editing process is the score function. This function allocates scores to individual records, based on which the records are given

a priority for manual editing. The records with the highest scores are eligible for editing first. Such scores are, in principle, the same as what is called the 'plausibility index' at Statistics Netherlands. The only difference is that, in the plausibility index, low values correspond to a high priority for manual editing, while the opposite is true for the regular score functions.

A score for a record can be composed of a number of different sub-scores or local scores. These are often separate scores for each of the major variables. These scores provide an indication of the expected effect of the editing of those variables for the estimates of the important target parameters, such as the totals of those variables and developments in those totals. A local score for a variable *j* in a record *i* generally has the following form,

$$s_{ij} = importance_{ij} \times risk_{ij}$$

The risk factor is determined by comparing the raw value of the variable with a so-called reference value. The reference value gives an indication of the value that could be expected, and is determined based on information from sources other than the current data set, such as a previous version of the same survey, other surveys or registers with similar variables. The degree to which a value deviates from the reference value determines the risk. The risk is high if the deviation is large; the raw value could then possibly be an error and, in that case, could also lead to a large correction. If the deviation is small, there is no reason to assume that the value could be an error. Moreover, if this were the case, then the correction would probably be small. The importance factor demonstrates how much the record contributes to the estimate of the publication figure. This factor is mainly related to the size of a company; a small correction in terms of percentage in the value of a large company could still have a substantial impact on a publication figure.

The local scores described above are related to the estimate for a target parameter. These scores are therefore also called estimator-related scores. Another type of scores is based on violations of edit rules (edits), such as the number of errors or the number of empty fields that should not be empty (partial non-response), which are both indicators of aspects of the quality of a record. This last type of scores is called edit-related scores. Section 4.3.3 demonstrates that edit-related and estimator-related scores can be combined into a single record score.

The following steps must be taken to implement a selective editing strategy:

- Defining local scores based on available reference values that approximate the expected values as closely as possible. Section 4.3.2 discusses several frequently used local score functions.
- Combining the local scores into a record score or global score. This is discussed in Section 4.3.3.
- Establishing a limit value for the record scores that can be used to select the records to be manually edited. The determination of the limit value is discussed in Section 4.3.4. The other records will be edited automatically. Automatic editing is discussed in Chapters 5 and 6.

4.3.2 Local score functions for totals and developments

The two most important target parameters in business statistics are totals of populations/subpopulations and developments within populations/subpopulations. This section discusses frequently used score functions for each of these target parameters. Most of the local score functions used in practice can be understood as variants of the functions discussed here, for which the importance or risk component is sometimes adapted to a specific situation.

To construct a score function that focuses on the effects of the editing on the estimate of totals, we first take a look at the usual estimator for the population total of a target variable y_j. This estimator can be written as

$$\hat{Y}_j = \sum_{i \in s} w_i \hat{y}_{ij} \ , \qquad (4.3.1)$$

where s is the set of responding units and w_i are weights that correct for unequal inclusion probabilities and non-response. The $\hat{y}_{ij}$ in (4.3.1) are edited values; in other words, they have gone through an editing process in which some of the raw values, say y_{ij}, were replaced by editors or by an automated process with better values $\hat{y}_{ij}$. The effect of the editing of a single record on the ultimate estimate can be expressed as the difference

$$d_{ij} = w_i (y_{ij} - \hat{y}_{ij}) . \qquad (4.3.2)$$

The variable d_{ij} contains the unknown corrected value of $\hat{y}_{ij}$ and therefore cannot be calculated. For this reason, $\hat{y}_{ij}$ is approximated by a reference value. The reference value serves as an assessment gauge for the quality of the raw value. The literature refers to the reference value also as the 'anticipated value'. The list below contains frequently used sources for reference values:

- Edited data from the same company from an earlier version the same survey, possibly multiplied by an estimate of the development between the current and previous observation. This source is much more important for short-term statistics than for annual statistics, because the overlap between the samples for consecutive periods is much larger in this context.
- Data from the same company from another survey or a registration. For Structural Business Statistics, data from, for example, Short Term Statistics, can be used, and tax data can be used for both.
- Data about a homogeneous subgroup of similar companies. For Structural Business Statistics, for example, the median of the edited data from a previous period in the same subdomain is used. Subdomain are often formed by SBI categories and size classes, or combinations thereof.

Besides the unknown corrected value, (4.3.2) also contains a weight w_i that is not yet known. Because the weights w_i correct for both unequal inclusion probabilities and non-response, they can only be calculated if the non-response is known,

therefore after the period in which the data is collected. However, the editing already begins during this period. A score function for selective editing therefore cannot make use of these weights. As a solution, it is common to approximate the weights w_i using the 'initial weights', say v_i, that only compensate for the unequal inclusion probabilities and which can already be calculated once the sample design is known: namely, as the inverse of these inclusion probabilities. If an estimate of the expected non-response can be made in advance, this can be used when determining the weights. Using the reference value and the initial weights, the effect of the editing on the estimate of the total can be quantified by the score function

$$s_{ij} = v_i \left| y_{ij} - \tilde{y}_{ij} \right| = v_i \tilde{y}_{ij} \times \left| y_{ij} - \tilde{y}_{ij} \right| / \tilde{y}_{ij} = b_{ij} \times r_{ij}, \tag{4.3.3}$$

where $\tilde{y}_{ij}$ is the reference value. As (4.3.3) demonstrates, this score function can be described as the product of an 'importance factor' b_{ij} and a 'risk factor' r_{ij}. The importance factor is the share of the record i in the total estimate based on the reference values, and the risk factor is the absolute value of the relative deviation of the observed value compared to the reference value. The risk r_{ij} represents the expected extent of change due to the editing. The relative importance $b_{ij} / \sum_i b_{ij}$ is often used instead of the importance b_{ij}. Because $\sum_i b_{ij} = \sum_i v_i \tilde{y}_{ij} = \tilde{Y}_j$, the resulting score $s'_{ij} = s_{ij} / \tilde{Y}_j$ can be understood as a scaled version of s_{ij}; by dividing by an estimate of the total (based on the reference values), the score becomes independent of the measurement unit. This scaling makes the scores for different variables easier to compare, which offers advantages when local scores are combined into a record score (see section 4.3.3).

Please note that the score defined in this manner takes on higher values as the risk increases, *and* as the importance rises, thus as the probability of an influential error increases the record becomes eligible for manual editing earlier in the process. In Structural Business Statistics at Statistics Netherlands, the scores are transformed into scores on a scale of 1 to 10, in which 10 stands for a very plausible value of y_{ij} and 1 for a very implausible value. After this transformation, the scores are called 'plausibility indicators'.

Another known score function is obtained by basing the risk factor on the ratio between the raw value and the reference value instead of the absolute difference as in (4.3.3). This risk factor, proposed by Hidiroglou and Berthelot (1986) is defined as follows:

$$r_{ij} = \max\left(\frac{\tilde{y}_{ij}}{y_{ij}}, \frac{y_{ij}}{\tilde{y}_{ij}} \right) - 1. \tag{4.3.4}$$

Because of this definition, upwards multiplicative deviations of the reference value count just as heavily as downwards multiplicative deviations, and the minimum value is 0, for $y_{ij} = \tilde{y}_{ij}$.

Sometimes, the ratios between variables in a record are more suitable for detecting deviating values than the separate variables themselves. Examples of this are the ratio between a company's turnover and its number of employees or the ratio between the price of a house and the number of square metres. The turnover per employee and the price per square metre show much less fluctuation than the turnover and the price. It is therefore easier to distinguish between deviating values, except for if the numerator and the denominator deviate in the same direction. Score functions based on ratios can be obtained by replacing y_{ij} and $\tilde{y}_{ij}$ in the risk factor in (4.3.3) or (4.3.4) by the raw value and the reference value of the ratio respectively.

For some statistics, such as short-term statistics, the most prominent target parameters are developments for populations/subpopulations. In these cases, it is common to choose a score function that is geared towards detecting companies with deviating developments. The development in a target variable y_j between the current time t and a previous time $t-1$, for a company i is $\hat{o}_{ij} = \hat{y}_{ij,t} / \hat{y}_{ij,t-1}$. We assume that the $t-1$ data has already been edited and that the intention of the score function to be calculated is solely to selectively edit the current data. The raw value of the development is therefore $o_{ij} = y_{ij,t} / \hat{y}_{ij,t-1}$. A risk factor in a score function will attempt to detect deviating values of the individual developments o_{ij} by comparing these with a reference value $\tilde{o}_{ij}$. For the reference value, Hidiroglou and Berthelot (1986) choose the median of the o_{ij} in a subdomain. The disadvantage of this is that editing can only be commenced when sufficient response has been received to determine this median. As an alternative, Latouche and Berthelot (1992) therefore select the median of the individual developments between $t-2$ and $t-1$, which is useful if the development between t and $t-1$ resembles that between $t-1$ and $t-2$. In the short-term statistics at Statistics Netherlands, the reference value is obtained by first determining a reference value for $\hat{y}_{ij,t}$ and then by calculating the reference value for the development as $\tilde{o}_{ij} = \tilde{y}_{ij,t} / \hat{y}_{ij,t-1}$. The reference value $\tilde{y}_{ij,t}$ is determined by extrapolation from $\hat{y}_{ij,t-1}$ with the help of a seasonal pattern estimated from earlier data. A reference value can be used to determine a risk factor, for which, in the case of developments, a multiplicative form is usually used, such as (4.3.4).

A score function can now be formed by multiplying r_{ij} by an importance factor; for this, Hidiroglou and Berthelot use the unweighted version of

$$b_{ij} = \left[\max\left(v_{i,t} y_{ij,t}, w_{i,t-1} \hat{y}_{ij,t-1}\right)\right]^c, \tag{4.3.5}$$

with $0 \leq c \leq 1$. Using the parameter c, the influence of the importance can be determined; the influence of the importance declines for lower values for c. Based on empirical research at Statistics Canada, Latouche and Berthelot suggest choosing the value 0.5 for c. The maximum function in (4.3.5) has the result that an error in $y_{ij,t}$ tends to overestimate rather than underestimate the importance. After all, a

reported value of $y_{ij,t}$ that is too low can never lead to an importance smaller than $\hat{y}_{ij,t-1}$, while an overly high value of $y_{ij,t}$ can, in principle, increase the importance to an unlimited extent. A scaled version of a score with an importance factor according to (4.3.5) can be obtained by dividing $v_{i,t}y_{ij,t}$ and $w_{i,t-1}\hat{y}_{ij,t-1}$ in (4.3.5) by estimates for their total, $\tilde{\tilde{Y}}_{j,t}$ and $\hat{Y}_{j,t-1}$ respectively. The estimated $t-1$ total is simply $\hat{Y}_{j,t-1} = \sum_i w_{i,t-1}\hat{y}_{ij,t-1}$. Because it is assumed that all the data is not yet available, the actual total must be approximated, for example, by an estimate based on reference values and initial weights: $\tilde{\tilde{Y}}_{j,t} = \sum_i v_{i,t}\tilde{\tilde{y}}_{ij,t}$. The scaled version of the importance factor can be written as

$$b'_{ij} = \left[\max\left(\frac{v_{i,t}y_{ij,t}}{\tilde{\tilde{Y}}_{j,t}}, \frac{w_{i,t-1}\hat{y}_{ij,t-1}}{\hat{Y}_{j,t-1}} \right) \right]^c . \qquad (4.3.6)$$

4.3.3 Combining local scores into a global score

A score at record level is needed to select – or not select – a record for manual editing. This global score combines the information from the local scores about the expected influence of the editing of different variables in the record into a single score that indicates the importance of the manual editing for the entire record.

When combining the scores, it is important that the order of magnitude of the scores is comparable, because otherwise the different scores will unintentionally be given a different weight in the global score. It is therefore common to scale the scores so that that they are easier to compare. One method for this is described in the previous section. Another method involves dividing the score by the standard deviation of the reference values, ($s_{ij} / \sigma(\tilde{\tilde{y}}_j)$); see Lawrence and McKenzie (2000). This second method has the advantage that deviations in variables with a large dispersion are given scores that are not as high, and therefore tend to be characterised less quickly as suspect than deviations in variables with a smaller dispersion.

Various methods have been proposed to combine the standardised scores into a global score. Often, the sum of the local scores is used (Latouche and Berthelot, 1992). Records with many deviating values consequently are given high scores and therefore high priority for manual editing. This is an advantage because editing multiple variables in the same record is relatively less work than editing a single variable in a record, certainly if additional contact takes place with the respondent for this purpose. The result of the method is that records with a high number of yet less strongly deviating values will tend to be manually edited earlier than records with a low number of strongly deviating values. If it is desirable that a strongly deviating value for a single variable in an otherwise non-suspect record should still be manually edited, the sum of the local scores is not a good criterion.

As an alternative for the sum of the local scores, Lawrence and McKenzie (2000) propose using the maximum of the scaled scores. The advantage of this is that the

method for each variable guarantees that deviating values above a certain limit will be inspected manually. The disadvantage of this safe strategy is that a distinction will no longer be made between records with a single serious deviation and records with many similarly serious deviations. As a compromise between the sum and the maximum, Farwell (2005) proposes using the Euclidian metric. These three proposals can be generalised to combining the local scores into a global score according to the so-called Minkowski metric (see Hedlin, 2008) given by

$$S_i^{(\alpha)} = \left(\sum_{j=1}^{J} s_{ij}^{\alpha} \right)^{1/\alpha}, \tag{4.3.7}$$

where J is the number of local scores. The parameter α in (4.3.7) determines the influence of high values of the local scores on the global score; this influence increases with α. For $\alpha = 1$, (4.3.7) is the sum of the local scores and for $\alpha = \infty$, (4.3.7) is equal to the maximum of the local scores, only the largest value still counts in this case. For $\alpha = 2$, (4.3.7) is the Euclidian metric.

For extensive questionnaires, such as the Structural Business Statistics, not all of the variables are equally important. Totals of turnover and numbers of employees are much more important than breakdowns of components of the operating costs. In such cases, weights are assigned to the local scores in the summation (4.3.7), and these weights express those differences in importance. In Structural Business Statistics, for example, content experts assign weights, for which the choice is between 0, 1, 10 and 100.

In formula (4.3.7), estimator-related score functions are combined. In addition, however, there are sometimes edit-related score functions, such as the number of hard errors or the number of improperly empty fields (partial non-response), which also say something about the quality of a record. The edit-related scores must then be added to the global score. This can be done by treating them in the same way as the estimator-related scores and, after the appropriate scaling and possibly with their own weights, adding them to the summation in (4.3.7). Another possibility is to first combine the edit-related scores with their own metric, and to add these combined scores to the combined estimator-related scores (see Section 4.4.2 for an example).

4.3.4 Determining the threshold value for the global score and pseudo-bias

The ultimate goal of a global score function is to select records for manual editing. If the editing can wait until after the observation period, the manual editing can take place according to the prioritisation of the global score until estimates of the most important target parameters no longer change substantially as a result (compare with Figure 3). Because manual editing is time consuming, this approach leads to unacceptably long turnaround times, especially for statistics with larger amounts of data and variables. To start manual editing during the data collection phase, it is necessary to take a decision, based on the score per record, without comparison with the scores of the other records, to edit or not edit the record manually. With this goal, a threshold value for the record score is determined such that records with a

score higher than the threshold value are edited manually and records with a score lower than the threshold value are edited automatically or not at all.

The usual method to determine a threshold value utilises a simulation study that investigates the effect of different threshold values, and therefore varying degrees of manual editing, on the bias in the most important target parameters. Such a simulation is based on a set of raw data and the associated data edited completely manually. This data must be comparable to the data on which the threshold values are going to be used. The usual choice for this is the data from an earlier version of the survey.

For the simulation study, global scores for the records with raw data are first calculated using the selected methods, then these records are put in order according to these scores. A simulation is then carried out in which only the first p% of the records are selected for manual editing. This is done by replacing for that first p% of the file the raw values by the edited values. We indicate the sub-file with the edited records by H_p.

Next, the difference is determined between the estimate of the total of a variable based on the p%-edited file and based on the fully edited file. The absolute value of the relative difference between these estimates is called the absolute pseudo-bias (Latouche and Berthelot, 1992), given by

$$D_j(p) = \frac{1}{\hat{Y}_j}\left|\sum_{i \in H_p} w_i(\hat{y}_{ij} - y_{ij})\right| \quad . \tag{4.3.8}$$

As (4.3.8) demonstrates, the absolute pseudo-bias is determined by the difference in the totals of the edited values and the non-edited values for the part of the records not selected for manual editing. If the result of the editing is that all errors (and *only* errors) are corrected, then (4.3.8) is the relative bias that arises because not all of the records are edited. Because it is not certain whether the editing reproduces the actual values, (4.3.8) is an approximation of this bias and is therefore called the pseudo-bias.

The pseudo-bias in the case of p%-editing can also be seen as an estimate of the gains in accuracy that can be achieved by also editing the rest ($1 - p$%) of the records. By calculating the pseudo-bias for a large number of different values of p, an impression is obtained of the gains in accuracy as a function of p. If the sorting of the records based on the score has the desired effect, then these gains will decline as p increases. For some value for p, it will then be decided that the remaining pseudo-bias is small enough and that it is not worthwhile to edit more records. The record score corresponding with this value for p is then the threshold value.

The pseudo-bias as described above is based on a comparison between manually edited data and raw data, and therefore assumes either that editing will take place manually or not at all. In many cases, however, automatic editing takes place instead of no editing whatsoever. If we assume that, in any case, automatic editing does not lead to more bias than no editing at all, the abovementioned pseudo-bias can be

understood as an upper limit for the pseudo-bias in situations in which automatic editing is used.

4.4 Examples

This section examines several practical examples of selective editing and parts of this process. We first address the construction of a plausibility index for short-term statistics (§ 4.4.1) and then provide a short description of the plausibility index of the survey Building Objects in Preparation (§ 4.4.2). These practical examples serve only as an illustration of the techniques, and therefore not all of the implementation details are addressed. More extensive explanations can be found in Van Duin, 2003 (plausibility index for short-term statistics), and Van der Loo and Pannekoek, 2007 (plausibility index of the survey Building Objects in Preparation).

4.4.1 Score function for Short Term Statistics

The most important variable in Short Term Statistics (STSs) is the turnover, and the most important target parameter is the development in this turnover between consecutive periods (months or quarters). In the standard production process for these statistics at Statistics Netherlands (IMPECT 2), selective editing is carried out for which the selection is determined exclusively by the variable 'turnover'.

The selection process makes use of variants of the importance and risk factors that were discussed in Section (4.3.2). The risk factor is the ratio of the observed turnover development between t and $t - 1$ and a reference value for this development:

$$r_{i,t} = \frac{o_{i,t}}{\tilde{o}_t}, \text{ with } o_{i,t} = y_{i,t} / \hat{y}_{i,t}.$$

Note that, in this definition of a risk factor, in contrast to that in (4.3.4), both values much greater than 1 and values much smaller than 1 are 'suspect'.

To determine the reference value, the median for a number of past years is calculated for the turnover development between $t - 1$ and t. For a monthly statistic, t and $t - 1$ are always the same months but from different years, and for a quarterly statistic, t and $t - 1$ are always the same quarters, but from different years. Next, the geometric mean is calculated over the years, of these developments from the past. The reference value is determined by multiplying this geometric mean by a correction factor for the difference in the number of workdays in $t - 1$ and t.

The importance factor is the scaled importance according to (4.3.6), therefore the maximum of the contribution to the $t - 1$ total and an estimate for the contribution to the current total. Instead of $t - 1$, however, we look here at $t - 2$, so

$$b_{i,t} = \max\left(\frac{v_{i,t}, y_{i,t}}{\tilde{Y}_t}, \frac{w_{i,t-2}, \hat{y}_{i,t-2}}{\hat{Y}_{t-2}} \right).$$

We look at $t - 2$ because, for the period $t - 1$, an approved total may not yet be available. The approximation for the current total $\tilde{Y}_t$ is obtained by multiplying the estimated total for $t - 2$ by the estimated turnover development according to the reference value as calculated above.

Using the above risk and importance factors $r_{i,t}$ and $b_{i,t}$, records are selected for interactive editing. The strategy followed here is different from that in the strategy described in §4.3.1. Instead of combining the risk and importance factors into a score and then applying the selection using a threshold value for this score, separate threshold values for the importance and risk factors are used here. This selection process is summarised in the following two steps:

1. If $b_{i,t} > b_{\min 1}$ then interactive editing is performed, independently of the value of $r_{i,t}$. The value of $b_{\min 1}$ is chosen such that a small number of very important companies is selected for which it is worthwhile to always have them checked by an editor. For the STSs, this only relates to a couple of variables, therefore a record can be checked quickly.

2. The following applies for the other records:

Only if $b_{i,t} > b_{\min 2}$ and ($r_{i,t} < r_{\min}$ of $r_{i,t} > r_{\max}$), then interactive editing is performed. Note that both $r_{i,t} < r_{\min}$ and $r_{i,t} > r_{\max}$ indicate a large risk.

In the second step, the same as for the previously discussed approach, if the risk *and* the importance are high, then interactive editing is performed. In contrast to the previously discussed approach, however, there are fewer 'compensation opportunities' here. In the score function approach, a record with a low risk can still be given a high score, and therefore be selected for interactive editing if the importance is high enough. This is not possible in the approach used here; if the risk falls in the interval $[r_{\min}, r_{\max}]$, the record is plausible and it is not edited interactively. In addition, if $b_{i,t} < b_{\min 2}$, then the record is unimportant and is not edited interactively, no matter how high the risk is.

4.4.2 Plausibility index for the survey Building Objects in Preparation

The quarterly survey Building Objects in Preparation (BOP) follows the development of the total construction value of new contracts at architectural firms in the Netherlands, and is used as a quick indicator for developments in the construction industry (see also Section 2.4.1 where the deductive corrections in the editing process of this survey are discussed). The budget of such a contract is the main variable for this survey. This is used to make estimates of the total budget for building objects in the classes defined by the combinations of type of building (residence, non-residence, combined-purpose building) and type of work (new construction, renovation). It is therefore of primary importance to correct errors in the reported budget. The budgets of the building objects show a very large dispersion, which makes it difficult to detect deviating values. The budget per

square metre, however, shows a much smaller dispersion. Strongly deviating values for the budget per square metre are therefore an indication for potentially incorrect details in the budget (or the number of square metres). Therefore, a risk factor was selected based on the deviation of the budget per square metre of a building object compared to the median for the class concerned. If, for a construction project i in class k, we indicate the budget by c_{ki} and the area in square metres by a_{ki} , then the square metre price can be defined as $x_{ki} = c_{ki}/a_{ki}$. The risk factor can be written as

$$r_{ik} = \max\left(\frac{\tilde{x}_k}{x_{ki}}, \frac{x_{ki}}{\tilde{x}_k}\right),$$

where $\tilde{x}_k$ is the median of the square metre price of the building objects in class k. This is a risk factor of the form (4.3.4) (only the constant –1 is omitted here, which does not have consequences for the order of the records in terms of their risk).

An importance factor must represent the importance of respondent i for the estimator of the total of the target variable. The total budget per class, Y_k, is estimated by

$$\hat{Y}_k = \sum_i w_{ki} y_{ki} = \sum_i w_{ki} a_{ki} x_{ki},$$

where w_{ki} is the raising weight. The importance of a record for the estimator of the total budget can therefore be expressed in the importance factor $b_{ki} = w_{ki} a_{ki}$, and the risk and importance factors can be combined into the score function

$$s_{ki}^{(Y)} = b_{ki} r_{ki}.$$

This score function per class is scaled by dividing it by the maximum per class.

In addition to looking for influential suspect values of the budget per square metre, the selective editing of the BOP also looks for hard errors. These can be values that fall outside the permitted range (such as percentages of residential area that are not between 0 and 100) or missing values in the variables Budget or Number of residences. They can also be conflicting values, such as stating several object types for a single building object or filling in a percentage (<100%) of residential area for a building object that is a residence. In total, 13 types of hard errors have been defined. The number of hard errors is a sign of the quality of the record, and a score function is also defined for this:

$$s_{ki}^{(E)} = \sum_{j=1}^{J} g_j^{(E)} E_j,$$

in which E_j =1 if a hard error of type E_j occurs, and otherwise 0. With the weights $g_j^{(E)}$, the relative importance of the different hard errors can be established. The weights are scaled such that $\sum_j g_j^{(E)} = 1$. As a result, the score for hard errors falls between 0 and 1. By combining the two score functions in a weighted manner, the final score function results:

$$s_{ki} = g_1 s_{ki}^{(E)} + g_2 s_{ki}^{(Y)}.$$

with $g_1 + g_2 = 1$.

This score function is an estimator-related score $s_{ki}^{(Y)}$ combined with an edit-related score $s_{ki}^{(E)}$, where the relative importance of these two components is represented by the weights g_2 and g_1 respectively.

5. Error localisation based on the Fellegi-Holt paradigm

5.1 Short description

With this method, a data file is checked record by record using predefined edit rules. If a record violates one or more edit rules, the method produces a number of fields that can be imputed so that no more rules are violated. The imputation itself is not part of the method.

When selecting the fields, the – generalised – Fellegi-Holt paradigm is assumed. This means that the smallest (weighted) number of fields is selected which will allow the record to be imputed consistently. Designating the fields to be imputed is called error localisation, and this can be performed in an automated manner. The edit rules can be both arithmetic (such as checking sums) and logical in nature (for example: if *gender* = man then *pregnant* = no). Combinations are also possible.

At Statistics Netherlands, software has been developed to perform this automated error localisation, in the form of SLICE/CherryPie.

5.2 Applicability

This method is intended to detect incorrectly filled in fields in a record. The method can be used on data files that have numerical, categorical or both data types. For numerical data, edit rules must consist of linear relationships between the variables (see 5.3.1). For categorical data, any relationship can be established between variables. It is essential that edit rules can be checked per record. For example, an edit rule for which the value in a field is compared with the average value for that field over the entire file is not a valid edit rule. This does mean that this error localisation method can be used before all the data has been received.

The generalised Fellegi-Holt paradigm can be used for every survey, even though it is not suitable for all types of errors. For some inconsistencies, such as unit errors (for example, unit of measure-errors), interchanged signs and interchanged columns, it is better to use deductive correction, such as described in Chapter 2. The most important difference between deductive correction and the method described here is that deductive correction makes use of the stated values in fields to localise errors, and the current method does not. If the value in a field can provide an indication about the error that has arisen (and therefore the solution), it is better to use deductive correction. In addition to the above examples, interchanged figures and comma errors are other examples of this type.

In error localisation according to the generalised Fellegi-Holt paradigm, no distinction is made between hard and soft edit rules: all rules are treated as hard edit rules. Hard edit rules are rules that are established by arithmetic or logical relationships, such as *turnover* = *profit* + *costs*. Hard edit rules define value combinations that are *certainly* wrong. Soft edit rules indicate whether a value, or

value combination, is *unlikely*, such as *costs* / *turnover* ≥ 0.6. In error localisation, all records that violate one or more edit rules are viewed as certainly wrong. If too many soft edit rules are defined, there is a danger of *over-editing*: the unjustified adaptation of correctly filled-in data. See, for example, Di Zio et al. (2005).

5.3 Detailed description

The description provided here focuses mainly on automated error localisation, in which a field choice is determined for each record on the fly. For surveys with few questions, based on the generalised Fellegi-Holt paradigm, a field choice can be established manually per combination of violations. This took place, for example, in the editing of the statistic Building Objects in Preparation (Van der Loo and Pannekoek, 2007), in which five variables play a role. If the number of variables and the relationships between them increase, the complexity of the error localisation problem rises quickly. For this reason, at Statistics Netherlands, software for error localisation has been developed in the form of SLICE (De Waal, 2005a). SLICE can be used to process large and complex surveys. SLICE is used, for example, when processing the data in Structural Business Statistics (De Jong, 2002).

This chapter is structured as follows. In Section 5.3.1, we describe the formulation of records and edit rules. In Section 5.3.2, we provide an overview of the attributes of the error localisation problem and its solution. To resolve the error localisation problem, it is necessary to derive (automatically) from explicitly defined edit rules the rules that logically follow from them. The techniques for this are described in Section 5.3.3. This – rather technical – section may be skipped when reading the document for the first time. The subsequent section (5.3.4) is dedicated to the solution as implemented by Statistics Netherlands: the branch-and-bound algorithm. This section is also rather technical and may also be skipped. Finally, we examine the SLICE/CherryPie software developed by Statistics Netherlands, which can be used to resolve error localisation problems.

5.3.1 Records and edit rules

A record is a row of fields or variables from a questionnaire. A record x can be represented as $x = (x_1, x_2, \ldots, x_n)$. The values that can be taken by variable x_i are called the *domain* D_i. Examples are x_i = gender with $D_i = \{man, woman\}$, x_i = number of residences with $D_i = \mathbf{N}$ or x_i = profit with $D_i = \mathbf{R}$. The total domain D, in which all possible records fall can be written as $D = (D_1, D_2, \ldots, D_n)$.

Edit rules indicate what conditions variables or variable combinations must satisfy in a data file per rule. Edit rules are often called editing rules or edits or edit checks. All edit rules must be checkable per record, and must therefore not depend on values in fields of other records.

The types of edit rules that can be used can be distinguished based on the types of data to which they relate:

- **Numerical data.** Can be checked based on linear relationships such as $turnover \geq 0$, or *profit* + *costs* = *turnover*. The general form of a linear edit rule is as follows:

$$\sum_{i=1}^{n} a_{ji} x_i \geq b_j \quad \text{or} \quad \sum_{i=1}^{n} a_{ji} x_i = b_j ,$$

  where j numbers the edit rules, a_{ji} are linear coefficients and b_j are constants. Note that rules such as *costs* / *turnover* ≥ 0.6 are also linear edit rules, because they can be written in the form *costs* – 0.6 · *turnover* ≥ 0.

- **Categorical data.** Any combination of categorical data can be ruled out. The rules are often written in if-then form, for example: **if** *gender* = man **then** *pregnant* = no.

- **Combinations of both.** These are also written in if-then form, for example: **if** *marital status* = married **then** *age* ≥ 16.

If an edit rule e explicitly relates to variable x_i, we say that x_i *occurs* in e. Conversely, we say that e *contains* x_i. Note that an edit rule establishes a subset of all possible records in D, for which all records in that subset contain at least one error (see also Section 5.3.3).

To resolve the error localisation problem, it is important that account is taken not only of the predefined rules, but also the rules that logically arise from them. Rules that are defined by the user are called *explicit edit rules*, and rules that are derived from them are known as *implicit edit rules*. For example, given the edit rules $x_1 > x_2$ and $x_2 > x_3$, then the implicit rule $x_1 > x_3$ is derived from this. It is not necessary (not even possible for linear rules) to generate all implicit edit rules. Fellegi and Holt (1976) demonstrated that it is sufficient to derive the so-called *essentially new* edit rules (see Section 5.3.3).

5.3.2 Error localisation

Error localisation involves designating one or more fields in a record, such that after adapting the content of these fields, the record no longer violates any edit rules. It is important to understand that it is not certain that the actual error (made by the respondent) will be found. An assumption is always made in designating the 'incorrect' values. The generalised Fellegi-Holt paradigm is based on such an assumption, and can be summarised as follows: if a record x violates one or more edit rules, we look for the set of fields G that satisfy:

(G1) The content of the fields $g \in G$ can be adapted, such that record x no longer violates any explicit or essentially new edit rules.

(G2) The value of $\sum_{g \in G} w(g)$ is minimised.

Here $w(g)$ are reliability weights for the fields g in G. A higher value for $w(g)$ means that field g is considered to be filled in better. Testing the reliability weights is not possible in the error localisation method. The validity of the selected reliability weights must therefore be investigated separately. See, for example, Hoogland and Smit (2008).

A special case arises when the same value is chosen for all weights $w(g)$, for example $w(g)=1$ for all g. In that case, we still have error localisation based on the original Fellegi-Holt paradigm. The set G then consists of the smallest set of fields with which the record can be consistently imputed. In that case, it can be proven (Fellegi and Holt, 1976) that G is given by the smallest possible set of variables that covers all violated explicit and essentially new edit rules. That means the smallest set of fields for which each field occurs in at least one of the violated explicit and essentially new rules. The assumption behind this is that errors are made randomly, and that the largest set of consistently completed fields is filled in truthfully.

A final option involves not making an extra assumption about the best solution for the error localisation problem, and selecting a random solution from all sets of fields that satisfy the requirement (G1).

Even if the generalised Fellegi-Holt paradigm is used, the error localisation problem can have multiple solutions. To generate a unique solution in that case, use can therefore be made of a hierarchical combination of selection principles. For example: (1) Generate the solutions $G_1, G_2, \ldots, G_m$ according to the generalised Fellegi-Holt paradigm. (2) If the solution is not unique $(m>1)$, then select a random solution from the m possibilities. Another method could be: (1) Generate the solutions $G_1, G_2, \ldots, G_m$ according to generalised Fellegi-Holt paradigm. (2) If the solution is not unique $(m>1)$, then select the solution with the smallest number of fields. (3) If the solution is still not unique, then select one randomly from remaining solutions. See also Stoop (2003) for more selection mechanisms.

Assuming the generalised Fellegi-Holt paradigm, there are different algorithms to find the possible solutions $G_1, G_2, \ldots, G_m$. The method implemented at Statistics Netherlands in CherryPie (a part of SLICE) is based on the so-called *branch-and-bound* algorithm (De Waal, 2003; 2008). Summarised briefly, this algorithm is used to test whether each relevant combination of fields can satisfy the requirement (G1). The relevant combinations can be covered using a binary tree. Then, based on the selection principles or a combination thereof, a choice can be made from the possible solutions. This algorithm is described in Section 5.3.4. The algorithm uses a record and a set of edit rules as input. The output consists of a set of fields that can be consistently imputed. The complexity (the extent to which the execution time of the algorithm increases with the input) of the *branch-and-bound* algorithm is rather high. First, constructing the tree has an asymptotic (maximum) complexity of $\mathrm{O}(2^n)$ in the number of variables. This means: each extra variable that occurs in an

edit rule can double the execution time. During the construction of the tree, variables from edit rules must be eliminated in each step. For categorical variables, this is a problem with complexity $O(2^{k_s})$, where k_s is the number of edit rules that x_s contains. The exact execution time therefore rises quickly with the number of variables and the number of edit rules. Because the execution time rises so quickly, a decision was made to use a time limit of several minutes for the implementation of this method for Structural Business Statistics. Records for which no solution is found after this amount of time are then edited manually.

Measures can be taken to reduce the execution time, namely by keeping n and/or k as small as possible. First, a data file can be pre-processed, so that as many as possible deductive corrections have already been applied. The *branch-and-bound* algorithm finishes more quickly when fewer rules have been violated. Second, the columns of the files that are not related to one another by correction rules can be treated as separate blocks (reduction of n). Third, the binary tree in SLICE is built cleverly: the order in which it is constructed was chosen such that solution can usually be found quickly, and branches that do not offer any solutions, or any solutions better than those found previously, are broken off (De Waal, 2005b; Daalmans, 2000). Finally, in electronic observation, account can be taken of the editing process by building editing rules into the questionnaire. By building hard edit rules into, for example, web forms, the number of edit rules violated is reduced for the later editing process. By making smart choices about the edit rules to be built in, the number of variables that the branch-and-bound algorithm must take account of can be decreased (see also Van der Loo, 2008).

5.3.3 Eliminating variables

It is possible to use logical or arithmetic calculations to generate implicit edit rules from a given number of explicit rules. For example, take a look at the following two linear edit rules:

e_1: *costs* + *profit* – *turnover* = 0

e_2: *costs* – 0.6 · *turnover* ≥ 0

By solving the costs from e_1, and substituting this in e_2, we obtain

e_3 : 0.4 · *turnover* – *profit* ≥ 0

The new rule e_3 does not contain the variable *costs*, while e_1 and e_2 do. We say that the variable *costs* has been eliminated. The general procedure to derive linear edit rules is called Fourier-Motzkin elimination (see, for example, De Waal, 2003, page 46). The method consists of solving a variable from one of the linear equations/inequalities, after which the solution is substituted into the other edit rules, taking account of the signs of the inequalities.

There is another procedure for categorical (logical) edit rules. For this purpose, we first define the *normal form* for categorical edit rules. Each edit rule for categorical

variables can be written as a combination of subsets of the domains $D_i, 1 \leq i \leq n$, namely:

$e_j = (F_1^j, F_2^j, \ldots, F_n^j)$, where each $F_i^j \subseteq D_i$.

The subsets are defined such that, if a record $x \in e_j$, then x violates edit rule e_j.

Take, as an example, a file with the fields $x_1 = \textit{marital status}$, $x_2 = \textit{age}$ and $x_3 =$ *relationship to head of household*. The domains pertaining to these variables are indicated by:

$D_1 =$ {*married, unmarried, widowed, divorced*}

$D_2 = \{<16, \geq 16\}$,

$D_3 = \{$*spouse, child, other*$\}$

The edit rule that says that someone younger than 16 years of age cannot be married looks as follows in normal form:

$e_1 = (\{$*married*$\}, \{< 16\}, \{$*spouse, child, other*$\})$

The edit rule that says that someone who is not married cannot be a spouse is represented in this notation as

$e_2 =$ ({*unmarried, widowed, divorced*}, {< 16, ≥ 16}, {*spouse*})

In other words, an edit rule establishes a subset of the total domain D in which all records in that subset contain at least one error. An edit rule e_j contains exactly those variables x_i for which $F_i^j \subset D_i$ ($F_i^j \neq D_i$). Therefore, rule e_1 in the example contains the variables *marital status* and *age*, and rule e_2 contains *marital status* and *relationship to head of household*.

In view of the two edit rules from the example, it is intuitively clear that someone who is younger than 16 years of age cannot be a spouse or a head of household. This rule can indeed be formally derived from e_1 and e_2. The general procedure proceeds as follows. Given two edit rules e_j and e_k, a new implied edit rule $F_s(j,k)$ can be formed by means of the operation

$$F_s(j,k) = (F_1^j \cap F_1^k, F_2^j \cap F_2^k, \ldots, F_{s-1}^j \cap F_{s-1}^k, F_s^j \cup F_s^k, F_{s+1}^j \cap F_{s+1}^k, \ldots, F_n^j \cap F_n^k),$$

where $F_s(j,k) = \varnothing$ if $F_i^j \cap F_i^k = \varnothing$ for one or more of the fields x_i, with $i \neq s$. The rules e_j and e_k are called *generating edit rules* and x_s is known as the *generating field*. It is easy to see that $F_s(j,k)$ is indeed an edit rule. Namely, if $x \in F_s(j,k)$, then it follows from the definition that $x \in e_j$, and/or $x \in e_k$. Because $F_s(j,k)$ is an edit rule, it directly follows that implied edit rules can be used to

produce new implied rules. The formula for $F_s(j,k)$ can therefore be simply generalised to $F_s(E)$ with E a set of edit rules. It is possible that the resulting edit rule no longer contains the variable x_s. This happens, for example, when $F_s^j \cup F_s^k = D_s$, and is called the *elimination* of x_s. The result is that, if a record violates the rule $F_s(j,k)$, this record, without any adaptation of x_s can be corrected such that both rules e_j and e_k are satisfied.

In the example, the variable *marital status* can be eliminated by forming the rule $F_1(1,2)$ in the following way:

$$F_1(1,2) = (\{\textit{married, unmarried, widowed, divorced}\}, \{< 16\}, \{\textit{spouse}\})$$

This rule can indeed be interpreted as: someone younger than 16 years of age cannot be the spouse of the head of the household. A record $x \in F_1(1,2)$ cannot be corrected for e_1 and e_2 by adapting the variable *marital status*. Namely, if, in x, the value *married* is substituted for *marital status*, x violates both e_1 and e_2; if another value is substituted, x violates rule e_2.

Edit rules that contain both categorical and numerical data can be written in a general form. To this end, we first write a record as $x = (v,y) = (v_1, v_2, \ldots, v_n, y_{n+1}, y_{n+2}, \ldots, y_{n+m})$, with categorical variables v_i and numerical variables y_i. The general form for edit rules is then provided by combining the normal form of categorical rules with linear edit rules using an if-then statement:

$$e_j : \textbf{IF } v \in F^j \textbf{ THEN } y \in \{y : a_{j\bullet} \cdot y \geq b_j\},$$

where F^j is a subset of all possible combinations of categorical variables. The THEN condition is a linear condition for y with $a_{j\bullet} = (a_{j1}, a_{j2}, \ldots, a_{jm})$. Note that linear equations can also be represented in this notation, because each linear equation can be written as two linear inequalities.

For a numerical variable y_s, these rules can be combined into an implied edit rule as follows:

$$F_s(j,k) = \textbf{IF } v \in F^j \cap F^k \textbf{ THEN } y \in \{y : \tilde{a} \cdot y \geq \tilde{b}\},$$

where $\tilde{a}$ and $\tilde{b}$ are the linear coefficients and constant which are obtained by eliminating y_s from the THEN conditions of e_j and e_k using Fourier-Motzkin elimination. To resolve the error localisation problem, it is not necessary to generate implicit rules from such general edit rules in which the generating field is a categorical variable. The branch-and-bound algorithm is constructed such that categorical variables are only dealt with after all numerical variables have been processed (see also the following section).

As stated above, it is important to know not just the explicit edit rules, but also the implicit ones, for the error localisation problem. In general, the number of implicit edit rules can be extremely large or even infinite. However, Fellegi and Holt (1976) prove that it is sufficient to know a finite number of *essentially new* edit rules. An implicit edit rule $F_s(j,k)$ is an essentially new edit rule if

(E1) x_s does not occur in $F_s(j,k)$ and,

(E2) there is no edit rule of which $F_s(j,k)$ is a subset.

The first requirement says that edit rules are essentially new ones if a variable from the generating rules is eliminated. The second requirement establishes that redundant edit rules are not essentially new. Note, once again, that e_j and e_k can also be implicit rules.

5.3.4 The branch-and-bound algorithm

If a record $x = (x_1, x_2, \ldots, x_n)$ violates one or more edit rules, a binary tree is used to find the set of possible error patterns. A binary tree is a frequently used data structure from computer science and is composed of *nodes* that are linked using *directed edges*, or *arrows*. There is a unique starting node, which is called the *root*. Two edges originate from the root, which connects the root node with two nodes, which are called *children*. Each node in the tree has a maximum of two children: the left child and the right child. Each child has exactly one parent. A node that has no children is called a *leaf*, and is found at the end of the tree.

A set of edit rules and a single variable are associated with each node, except for the root. The root contains all explicit edit rules, and no variable. The tree is constructed from the root by treating the candidate variables $x_1, x_2, \ldots, x_n$ one by one, as follows. Select variable x_1. In the left child of the root, it is assumed that x_1 contains the correct value, and in the right child, it is assumed that x_1 contains an error. Next, a set of correction rules is generated for the left child and the right child. For the left child, the correction rules are copied from the parent, and the value for x_1 is substituted from the record in those rules. The rules that remain must be valid for the non-selected variables $x_2, x_3, \ldots, x_n$ if x_1 is not adapted. After substituting the value in x_1, some editing rules can produce internal contradictions (for example, '$0 \geq 1$'). In that case, the children of this node cannot result in a solution of the localisation problem and this branch is broken off. If there is no internal contradiction, the branch can be continued. A situation may arise in which the set of edit rules contains tautologies, such as '1 = 1'. These rules can be eliminated because they do not contain any variables. For the right child, the variable x_1 is eliminated from the edit rules of the parent, using the methods from the previous section. The resulting set of edit rules in the right child are the edit rules that the variables $x_2, x_3, \ldots, x_n$ must satisfy, whatever value is substituted for x_1. Next, the

tree is continued by selecting x_2, and generating a left child and a right child for each child, as above. This continues until all variables $x_1, x_2, \ldots, x_n$ have been selected. The leaves that ultimately have variable x_n and for which the related set of edit rules does not have internal conflicts correspond to a solution G for the localisation problem that satisfies requirement (G1). It can also be proven that this procedure finds precisely all of the solutions, see theorems 1 and 2 from De Waal and Quere (2003). Each solution is given by moving along the unique path from the root to the leaf and keeping track of which variables are fixed and which have been eliminated. After this, one of the possible solutions must be selected using one of the selection principles stated earlier. It is not necessary to find all of the possible solutions that satisfy (G1). Namely, if, in one of the branches, the sum of the reliability weights of the eliminated variables is larger than that of a solution found earlier, this branch does not need to be continued. By continually retaining the solution with the smallest sum to reliability weights, it becomes more efficient to search for solutions that satisfy both the requirements (G1) and (G2).

It should also be noted that, above, it was assumed that a value was substituted for all fields $x_1, x_2, \ldots, x_n$ for the record concerned. If item non-response occurs, the empty fields can be eliminated in the original set of edit rules, because they must still be imputed in any case. For the other variables and rules, the tree can be constructed as described above.

The algorithm described above is a basic procedure. In practice (SLICE), additional adaptations were made, and we will discuss some of them here. First, the numerical variables were processed earlier than the categorical variables to prevent several technical difficulties in the elimination of variables (De Waal, 2005b). Second, for each record, an attempt can be made to go through the variables in the most favourable order possible, to ensure that solutions are found as quickly as possible (see Daalmans, 2000; De Waal, 2005b). Third, in addition to reliability weights, the status '*locked*' can be assigned to variables. In this case, the algorithm looks for solutions for which the variable concerned is not adapted. See also De Jong (2002).

5.3.5 Software at Statistics Netherlands: SLICE/CherryPie

SLICE 1.6 has been available at Statistics Netherlands since 2007. We refer to De Waal (2005a,b) for a detailed description; we only provide a short overview here of the options offered by SLICE.

SLICE is the software library for automatic editing developed at Statistics Netherlands. The different functions of SLICE are included in modules. The module CherryPie is able to resolve error localisation problems based on the generalised Fellegi-Holt paradigm. CherryPie can work with both numerical and categorical data, and can deal with linear, categorical and combined edit rules, as described in Section 5.3.1. The rules can be drawn up in the CherryPie script language developed for this purpose, but there is also a module with which rules can be imported from Blaise. There is also an imputation module for numerical data, which can implement

simple imputation methods based on ratio estimators. After imputation, the records do not always satisfy all the edit rules, because this is not explicitly accounted for in the imputation method. For this reason, there is an extra module (AdaptValues) that adapts the imputed values. It is possible to use SLICE in combination with other software for imputation. De Jong (2002) provides an extensive overview of the options offered by SLICE.

SLICE itself does not have a graphical or other user interface, but consists of a library of routines in the form of a .dll (dynamically linked library) file that can be utilised from other programs. This setup makes SLICE very flexible in its use. There is also a demonstration program available (SLICEDemo, see Sluis, 2004) that allows SLICE functionality to be tested.

5.4 Example

As an example, we will elaborate a small part of the editing process for the statistic Building Objects in Preparation (BOP) (see Van der Loo and Pannekoek, 2007). In BOP, architectural firms are asked about new assignments. They are asked about such things as the type of building object $t \in \{r, c, o\}$, in which r stands for residence(s), c for combined-purpose buildings (part residence and part other use) and o for other buildings. Questions are also posed about the percentage of living area $p \in [0,100]$ (in the case of a combined-purpose building) and the number of residences $n \in \mathbf{N}$. It is obvious that, for the category 'other', both the percentage of living area and the number of residences must be equal to 0. In the notation of the previous sections, this leads to the following edit rules:

$$e_1 = (o, (0,100], \mathbf{N})$$
$$e_2 = (o, [0,100], \mathbf{N}^+).$$

Here, e_1 says that, for other buildings, the percentage cannot be larger than 0, and e_2 says that, for other buildings, the number of residences cannot be larger than 0. We select all reliability weights equal to 1. In this case, there are no essentially new implied edit rules. Check in particular that $F_1(1,2) \subset e_1$, to ensure that that this does not satisfy either of the requirements (E1) or (E2). Further, verify that $F_2(1,2) = e_2$ and $F_3(1,2) = e_1$, so that these rules also do not satisfy (E2). Now consider a record $r = (o, 10\%, 0)$. This record only violates explicit rule e_1. Rule e_1 contains the fields t and p such that two minimal covering sets are possible, namely $G_1 = \{t\}$ and $G_2 = \{p\}$. Finally, consider a record $r' = (o, 10\%, 3)$. This record violates both e_1 and e_2. The only variable that occurs in both violated edit rules is the type of building t, so that there is only one optimum solution, namely $G = \{t\}$.

5.5 Characteristics

We also note that the error detection method based on the Fellegi-Holt paradigm is suitable for parallel processing by several servers. As all records are processed

independently, the processing time scales virtually in a linear fashion with the number of servers that can be deployed.

5.6 Quality indicators

The method works better if the errors actually made are detected. Using simulations, an impression can be obtained of whether this is indeed the case. It is possible, for example, to introduce realistic errors into a ‘perfect’ data file to determine under which conditions they are found using SLICE.

A second aspect can be the efficiency with which the branch-and-bound algorithm finds solutions for the error localisation problem. This can be checked in SLICE more or less by adapting reliability weights, or by designating variables as ‘locked’.

6. Error localisation with the Nearest-neighbour Imputation Methodology

6.1 Short description

The Nearest-neighbour Imputation Methodology (NIM) is an alternative method for automatic error localisation at record level. In contrast to the method from Chapter 5, the NIM is not based on the Fellegi-Holt paradigm, but on a principle derived from this. The NIM determines not only a solution to the error localisation problem – in other words, a set of fields that can be imputed to ensure that all edit rules are satisfied – but also the values to be imputed. In this regard, one can view this method also as an imputation method. In fact, the NIM is an extension of hot-deck donor imputation based on a distance function (see Chapter 6 in the theme *Imputation*), intended for the situation in which the data may still contain errors.

For each record that does not satisfy all the edit rules, the NIM draws up a list of donor records that (according to some distance function) closely resemble the record to be imputed. Using the donor records, the NIM determines ways to indicate errors in the record, so that the incorrect fields can be imputed with the accompanying values from a donor record, in such a way that all the edit rules are satisfied. Finally, the NIM selects the best of all proposed imputed versions of the record, according to a criterion that is explained in Section 6.3.

To apply the NIM, software called CANCEIS (CANadian Census Edit & Imputation System) developed by Statistics Canada is available at Statistics Netherlands.

6.2 Applicability

The NIM was developed at Statistics Canada for a single goal: the detection and correction of the census taken every five years (see, for example, Bankier et al., 1994). This is evident from several characteristics of the method:

- The NIM is able to process extremely large data sets quickly. An important condition, however, is that sufficient error-free donor records are available. This is exactly the situation that arises in a census: millions of records of which most of them contain no errors. The NIM is not a suitable method to use in a situation where almost all the records contain errors. In that case, the same records would be used as donor records repeatedly.
- The NIM can process both numerical and categorical data, and also a combination of both. The method is, however, mainly suitable for data sets with mainly categorical variables (and possibly a few numerical variables), such as in a census. The method is not suitable for completely numerical data sets that must satisfy linear equations, such as in Structural Business Statistics. In that case, it is nearly impossible to find a suitable donor from which a record can be imputed to satisfy the linear equations.

- The NIM uses the statistical attributes of the set of donor records as an approximation for the statistical attributes of the entire population. The method is therefore initially intended for statistics based on a complete count, such as the census. Data obtained from a random sample generally only provide a correct reflection of the entire population if raising weights are used. This is not possible with the current form of the NIM.

6.3 Detailed description

6.3.1 Records and edit rules

Just as in Chapter 5, we assume a data file with records from n fields, which we notate as $x=(x_1,\ldots,x_n)$. As stated in Section 6.2, a record may contain both categorical and numerical fields for the NIM.

For the localisation of errors, we use edit rules that indicate which values and combinations of values are not permitted. The implementation of the NIM in the current version of CANCEIS assumes that all edit rules have the following general form:

$$\text{if } (\ \Delta_1 \text{ and } \Delta_2 \text{ and } (\cdots) \text{ and } \Delta_S\ ) \textbf{ then } \varnothing . \qquad (6.3.1)$$

In numerical fields, each Δ_s represents a linear proposition of the form

$$\Delta_s: \quad a_{s1}x_1 + \cdots + a_{sn}x_n \lhd b_s ,$$

in which one of the operators $<,>,\leq,\geq,=,\neq$ must be substituted for the symbol $\lhd$. In categorical fields, each Δ_s has the form

$$\Delta_s: \quad x_i \in F_i^s \text{, for some } i \in \{1,\ldots,n\} ,$$

where F_i^s is a subset of the domain of x_i, analogous to Section 5.3.3. A record does not satisfy edit rule (6.3.1), and therefore contains an error, if all propositions $\Delta_1,\ldots,\Delta_S$ are evaluated as *true* when the values from the record are substituted.

To illustrate, we continually refer in this section to a small example with four fields: $x=(x_1,x_2,x_3,x_4)=$(*Age*, *Income*, *Marital Status*, *Relationship to Head of Household*). The first two fields are numerical with the non-negative integers as the domain. The last two fields are categorical. Variable x_3 has the possible values of *Married*, *Unmarried*, *Widowed* and *Divorced*. Variable x_4 has the possible values of *Spouse*, *Child* and *Other*.

In this example, there are three edit rules that the records must satisfy. Written in words, these rules are as follows:

1. People younger than 18 years of age cannot be or have been married.
2. People younger than 12 years of age do not have an income above 0 euros.
3. People who are not married cannot be the spouse of the head of the household.

We write the three edit rules as follows in the general form (6.3.1):

1. **if** ($x_1 < 18$ and $x_3 \in \{Married, Widowed, Divorced\}$) **then** $\varnothing$.
2. **if** ($x_1 < 12$ and $x_2 > 0$) **then** $\varnothing$.
3. **if** ($x_3 \in \{Unmarried, Widowed, Divorced\}$ and $x_4 \in \{Spouse\}$) **then** $\varnothing$.

Readers can work out for themselves that these two formulations describe the same rules.

6.3.2 Donor selection

To determine the extent to which two records resemble one another, we define a global distance function. The distance between the records $x^{(1)} = (x_1^{(1)}, \ldots, x_n^{(1)})$ and $x^{(2)} = (x_1^{(2)}, \ldots, x_n^{(2)})$ is

$$D(x^{(1)}, x^{(2)}) = \sum_{i=1}^{n} w_i D_i(x_i^{(1)}, x_i^{(2)}), \tag{6.3.2}$$

where $w_i \geq 0$ is the weight of variable x_i, and $D_i(x_i^{(1)}, x_i^{(2)})$ the distance between the values $x_i^{(1)}$ and $x_i^{(2)}$. For each variable, a local distance function is chosen, with the only conditions that $0 \leq D_i(x_i^{(1)}, x_i^{(2)}) \leq 1$ and that $D_i(x_i^{(1)}, x_i^{(2)}) = 0$ if $x_i^{(1)} = x_i^{(2)}$, and a weight that expresses the importance of the variable. A higher value of w_i means that variable x_i has a greater influence on the distance function. To eliminate a variable from (6.3.2), we select $w_i = 0$.

In the first step of the NIM, all records in the data file are checked using the edit rules selected by the user. Records that violate at least one edit rule apparently contain errors, and are subjected to automatic error localisation in the second step. All other records are placed in the so-called *donor pool*, in other words, the set of potential donor records. To use the NIM successfully, the donor pool must contain the vast majority of the records from the data file (see section 6.2).

The records that were designated for error localisation in the first step of the NIM are dealt with one by one in the second step. Given a record with errors $x^{(F)}$, the NIM looks for records $x^{(D)}$ in the donor pool with the smallest possible distance $D(x^{(F)}, x^{(D)})$. To keep the calculation time low, not all of the records from the donor pool are examined, but only the records that are located close to $x^{(F)}$ in the original data file. The underlying assumption is that the data file is sorted in such a way that records that are located close together are more alike than records that are located far apart. In the case of the Canadian census, it is common, for example, to sort the data file based on geographic attributes. The N_D records with the smallest distance $D(x^{(F)}, x^{(D)})$ are retained as potential donors, with N_D as an adjustable parameter.

In the example with four variables from section 6.3.1,

$$x^{(F)} = (x_1^{(F)} = 9, x_2^{(F)} = 25000, x_3^{(F)} = Unmarried, x_4^{(F)} = Spouse)$$

is a record that does not satisfy all the edit rules, for this record violates both the second and the third rule. An example of a record that does satisfy all the edit rules, and which is therefore suitable as a donor record, is

$$x^{(D)} = (x_1^{(D)} = 8, x_2^{(D)} = 0, x_3^{(D)} = Unmarried, x_4^{(D)} = Child)\,.$$

6.3.3 Generating imputation actions

After the potential donors for record $x^{(F)}$ are selected from the donor pool, the NIM tries to resolve the errors in $x^{(F)}$ by replacing some values by the associated values from a donor record $x^{(D)}$. Adopting values from a donor in another record is called an *imputation action*. We notate an imputation action formally as $I = (x^{(F)}, x^{(D)}, \delta)$, where δ presents a vector of binary variables, $\delta = (\delta_1, \ldots, \delta_n)$, with $\delta_i = 1$ if the value $x_i^{(F)}$ is replaced by $x_i^{(D)}$, and otherwise $\delta_i = 0$.

The result of the imputation action $I = (x^{(F)}, x^{(D)}, \delta)$ is an adapted record $x^{(A)} = (x_1^{(A)}, \ldots, x_n^{(A)})$, with

$$x_i^{(A)} = \delta_i x_i^{(D)} + (1 - \delta_i) x_i^{(F)}, \qquad i = 1, \ldots, n\,.$$

It is clear that we can disregard variables for which $x_i^{(F)} = x_i^{(D)}$ when generating imputation actions.

An imputation action is referred to as *feasible* if it produces an adapted record $x^{(A)}$ that satisfies all the edit rules. The NIM finds all feasible imputation actions that are generated by the N_D potential donors. Oftentimes, a donor record generates multiple feasible imputation actions.

In the example discussed previously, we obtain a feasible imputation action for $x^{(F)}$ by adopting the values of x_2 and x_4 from $x^{(D)}$. The adapted record is in that case

$$x^{(A)} = (x_1^{(A)} = 9, x_2^{(A)} = 0, x_3^{(A)} = Unmarried, x_4^{(A)} = Child)$$

and it is simple to establish that this record satisfies the three edit rules. In formal notation, we write this imputation action as $I = (x^{(F)}, x^{(D)}, \delta = (0,1,0,1))$.

Because $x_3^{(F)} = x_3^{(D)} = Unmarried$, in this example, only the three variables x_1, x_2 and x_4 can be used to generate useful imputation actions. In total, there are therefore $2^3 - 1 = 7$ possible imputation actions. Only two of these imputation actions are feasible; the only feasible imputation action in addition to the one previously stated imputes the variables x_1, x_2 and x_4, with the result:

$$x^{(A)} = (x_1^{(A)} = 8, x_2^{(A)} = 0, x_3^{(A)} = Unmarried, x_4^{(A)} = Child)\,.$$

6.3.4 Selecting from feasible imputation actions

In the example from Section 6.3.3, the following is true for the last imputation action: $x^{(A)} = x^{(D)}$. In general, there is always an imputation action with this attribute (select $\delta_i = 1$ for all $i = 1, \ldots, n$), and it is by definition feasible. Finding a feasible imputation action is therefore very simple. However, the goal of the NIM is more ambitious, and involves finding the *best possible* feasible imputation action. By 'best possible', in NIM the feasible imputation action $I = (x^{(F)}, x^{(D)}, \delta)$ is meant, with the following two attributes:

(B1) $x^{(A)}$ resembles $x^{(F)}$ as much as possible.

(B2) $x^{(A)}$ resembles $x^{(D)}$ as much as possible.

On the one hand, it is desirable that a feasible imputation action changes as little as possible in the original record; that is the rationale behind attribute (B1), which brings to mind the Fellegi-Holt paradigm from Chapter 5. On the other hand, the adapted record is artificial; it is composed of two different records. We know that the combination of values in the adapted record is not in conflict with the edit rules, but it is possible that this combination of values is very rare in the population[4]. Such imputation actions are not very plausible. As the adapted record increasingly resembles the donor record, the plausibility of the adapted record increases, because it resembles an error-free record that was obtained naturally; this is the rationale behind attribute (B2).

As an example, we examine the following record that does not satisfy edit rule 3 from Section 6.3.1:

$$x^{(F)} = (x_1^{(F)} = 56, x_2^{(F)} = 30000, x_3^{(F)} = \mathit{Unmarried}, x_4^{(F)} = \mathit{Spouse}),$$

and two potential donor records:

$$x^{(D,1)} = (x_1^{(D,1)} = 59, x_2^{(D,1)} = 28000, x_3^{(D,1)} = \mathit{Married}, x_4^{(D,1)} = \mathit{Spouse}),$$

$$x^{(D,2)} = (x_1^{(D,2)} = 21, x_2^{(D,2)} = 30000, x_3^{(D,2)} = \mathit{Unmarried}, x_4^{(D,2)} = \mathit{Child}).$$

Two feasible imputation actions are: impute $x_3^{(D,1)}$ or impute $x_4^{(D,2)}$. The accompanying adapted records are:

$$x^{(A,1)} = (x_1^{(A,1)} = 56, x_2^{(A,1)} = 30000, x_3^{(A,1)} = \mathit{Married}, x_4^{(A,1)} = \mathit{Spouse}),$$

$$x^{(A,2)} = (x_1^{(A,2)} = 56, x_2^{(A,2)} = 30000, x_3^{(A,2)} = \mathit{Unmarried}, x_4^{(A,2)} = \mathit{Child}).$$

[4] Cf. the distinction between hard and soft edit rules (Section 5.2). In automatic error localisation based on the NIM, all edit rules are interpreted as hard rules, the same as in error localisation based on the Fellegi-Holt paradigm. For this reason, soft edit rules cannot be used in the NIM to identify unusual value combinations, as can be done in interactive editing.

The records $x^{(A,1)}$ and $x^{(A,2)}$ both satisfy the edit rules. In both cases, only one variable from the original record was changed. In the population, however, there will be far more 56-year-olds who are married to the head of the household that they are a part of, than 56-year-olds who are a child of the head of the household. Please note: donor record $x^{(D,2)}$ itself does not belong to a 56-year-old who is a child of the head of the household, but such a record arises if the values from $x^{(F)}$ and $x^{(D,2)}$ are combined.

For every feasible imputation action $I = (x^{(F)}, x^{(D)}, \delta)$, the NIM determines the following measure:

$$\mu(I) = \alpha D(x^{(F)}, x^{(A)}) + (1-\alpha) D(x^{(A)}, x^{(D)}) .$$

Here, $D(x^{(F)}, x^{(A)})$ and $D(x^{(A)}, x^{(D)})$ are defined by means of (6.3.2), and α is a parameter to be selected by the user with $1/2 < \alpha \leq 1$. The best possible imputation action is now defined as the imputation action with the smallest value of $\mu(I)$. The choice of α determines whether, and if so, to what extent, we are looking at the plausibility of the adapted record: with $\alpha = 1$, the NIM only looks at attribute (B1), with $\alpha < 1$, attribute (B2) also plays a role. In the detection and correction of the Canadian census, the values $\alpha = 0.75$ and $\alpha = 0.9$ have been used.

By considering $D(x^{(A)}, x^{(D)})$ in the assessment of feasible imputation actions, the hope is that the NIM is able to retain univariate and multivariate distributions in the population. An adapted record $x^{(A)}$ with some combination of values that is present in, say, 5% of all donor records, is expected to also have a small $D(x^{(A)}, x^{(D)})$ in approximately 5% of the cases. In the other 95% of the cases, $D(x^{(A)}, x^{(D)})$ is large and the associated feasible imputation action will probably not be selected. In this context, it is assumed that the donor pool is a good reflection of the population as a whole.

Suppose that, for the feasible imputation actions in a record, the smallest value of $\mu(I)$ is equal to $\mu_{\min}$. In the terminology of the NIM, a feasible imputation action is called a *near minimum change imputation action* (NMCIA) if it satisfies

$$\mu(I) \leq \gamma \mu_{\min} ,$$

where $\gamma \geq 1$ is a parameter to be selected by the user. For each record, only the NMCIAs are retained. One can select γ slightly larger than 1, because feasible imputation actions with $\mu(I)$ close to $\mu_{\min}$ are only minimally worse than the best possible imputation action. The retention of such imputation actions helps to prevent that the same donor records are used repeatedly for the imputation. The value of $\gamma = 1.1$ has been used for the census in Canada.

Finally, the NIM makes a random selection from the list with NMCIAs in a record. The associated adapted record $x^{(A)}$ replaces $x^{(F)}$ in the output. In this way, all records that do not satisfy the edit rules are dealt with one by one.

6.3.5 Software: CANCEIS

For use of the NIM at statistical bureaus, Statistics Canada has made the CANCEIS software available free of charge. CANCEIS is still undergoing development using the experiences from the Canadian census. The description below is based on CANCEIS version 4.5 from 2006, which is available at Statistics Netherlands. See also CANCEIS (2006).

CANCEIS consists of three modules. The first module analyses edit rules and functions only as support for the other modules. The *Derive Module* is used to perform derivations and deductive corrections using correction rules (see Chapter 2). Finally, the *Hotdeck Module* contains an implementation of the NIM. This implementation is an efficient algorithm used to search for feasible imputation actions; see Bankier (2006) for an extensive explanation of this algorithm.

The input of CANCEIS consists of a number of ASCII files, including the raw data file and a file with edit rules of the type (6.3.1). The edit rules must be formulated in the form of *Decision Logic Tables* (DLTs). A DLT consists of rows and columns. Each column except for the first one corresponds to an edit rule. The first column contains all propositions Δ_s which occur in the edit rules, and each row relates to the proposition in its first column. The inner part of a DLT is composed of the elements 'Y', 'N' and '–', which indicate whether and, if so, how a proposition occurs in a certain edit rule: 'Y' means that the proposition itself occurs, 'N' means that the negation of the proposition occurs, and '–' means that the proposition does not occur.

To illustrate, we write out the three edit rules from section 6.3.1 in a DLT:

```
                      1   2   3
x1 < 18               Y   -   -
x1 < 12               -   Y   -
x2 > 0                -   Y   -
x3 = Unmarried        N   -   -
x3 = Married          -   -   N
x4 = Spouse           -   -   Y
```

For more examples of DLTs, see CANCEIS (2006) and Scholtus (2008b).

6.4 Example

At Statistics Netherlands, CANCEIS was tested for use in the production of demographic statistics based on the Municipal Personal Records Database. The situation is somewhat comparable with the Canadian census: there is a more or less complete population file with a large number of records, in which errors occur sporadically. The NIM seems to be a suitable choice for this purpose, because sufficient donor records are available. See Pannekoek et al. (2008) and Scholtus (2008b) for more information about this application of CANCEIS.

6.5 Quality indicators

The quality of a method for automatic error localisation is initially determined by the extent to which incorrect fields are correctly identified. Because the NIM also contains an imputation method, the extent to which the imputations correspond to the actual values is also important. As such, there may be interest in both the quality of the individual imputations and in the extent to which the imputed data correctly represents certain population distributions. In practice, all these attributes can only be measured using simulations, in which known errors and missing values are put into a 'perfect' data file.

Another aspect of the quality of the NIM involves the efficiency of the search algorithm. Users themselves can influence the calculation time needed to a certain extent, by their choices for parameter N_D and for the number of records in the vicinity of $x^{(F)}$ that are examined during the search for potential donors (see Section 6.3.2).

7. Macro editing

7.1 Aggregate method

7.1.1 Short description

In the *aggregate method*, aggregates are first calculated, usually the publication figures. If the calculated aggregates clearly deviate from what was expected, for example, based on previous data, then a lower aggregation level is examined and the underlying records are checked and possibly corrected. For aggregates that deviate only a little from what is expected, further checks can be performed to determine whether an aggregate is correct. Aggregates can be incorrect due to influential errors or incorrect weights.

7.1.2 Applicability

The goal of the aggregate method is to approve publication figures. For this purpose, a lower aggregation level can be considered to determine the stability of the figures, for example, per size class. If levels or growth rates deviate from the expectation, then potential influential errors can be detected using score functions.

It is essential to look at aggregates, especially if the observed data is incomplete and therefore imputed or raised. In addition to problems with the microdata, there can also be problems with the imputation or weighting method. Influential errors can be missed or introduced during micro editing. The aggregate method is mainly useful if influential errors occur structurally in microedited data, or if there are structural problems with weights or imputations.

The conditions for the aggregate method are:

- Systematic errors (very clear and less clear errors) are eliminated during micro editing.
- Each record has an observation or imputation available, or there is a weight available for each observed record;
- There are not too many influential errors in the microedited data. Or, useful aggregates can be determined;
- There is a reference framework. Or, there is reference data or an editor has sufficient sector knowledge to assess aggregates.

A publication figure may be plausible, but that does not mean it is correct. There may still be influential errors in the data. It is therefore recommended to combine the aggregate method with the distribution method; see Section 7.2. The distribution method can also be used to determine whether there are still systematic or influential

errors in the data. Furthermore, outliers can be detected with the distribution method.

7.1.3 Detailed description

There are various reasons why publication figures deviate from the expectation.

- There may be influential measurement or processing errors in the data;
- There may be problems with the weight framework or the weight method;
- There may be unexpected developments which are in fact real.

To determine whether the microdata must be examined further, the relative deviation of an aggregate for variable y_j in period t can be calculated compared to a reference aggregate $\hat{Y}_j^s$:

$$\frac{\hat{Y}_j^t - \hat{Y}_j^s}{\hat{Y}_j^s}, \tag{7.1.1}$$

for which

$$\hat{Y}_j^t = \sum_{i=1}^{n} w_i^t y_{ij}^t \ . \tag{7.1.2}$$

The reference aggregate can be determined based on another source or the same source for a previous period s.

You can also search for a ratio between two related variables y_j and y_k. A ratio is, in principle, more stable and easier to interpret than the variables separately. The relative deviation of a ratio can be determined as follows:

$$\left(\frac{\hat{Y}_j^t}{\hat{Y}_k^t} - \frac{\hat{Y}_j^s}{\hat{Y}_k^s}\right) \Big/ \frac{\hat{Y}_j^s}{\hat{Y}_k^s} \ . \tag{7.1.3}$$

If aggregates or ratios deviate too much from the expectation, then it is advisable to re-examine the underlying data. This can be done using the distribution method; see Section 7.2. Score functions can also be used to detect possible influential errors; see Section 4.3.2. A big advantage is that the weights and aggregates for the reporting period are available at that time. Using score functions, the influence of a record can be adequately included in the error detection.

In (7.1.1) and (7.1.3), no account is taken of the sample variance of aggregates. If an estimate for the standard deviation of the difference in aggregates and ratios respectively is available, then the relative deviations below can be determined:

$$\frac{\hat{Y}_j^t - \hat{Y}_j^s}{s.d.(\hat{Y}_j^t - \hat{Y}_j^s)}, \tag{7.1.4}$$

$$\left(\frac{\hat{Y}_j^t}{\hat{Y}_k^t} - \frac{\hat{Y}_j^s}{\hat{Y}_k^s}\right) \Big/ s.d.\left(\frac{\hat{Y}_j^t}{\hat{Y}_k^t} - \frac{\hat{Y}_j^s}{\hat{Y}_k^s}\right). \qquad (7.1.5)$$

7.1.4 Example

Using a method developed at Statistics Netherlands, based on VAT turnover levels and developments, an estimate can be made of the turnover of the small and medium-sized businesses in the Retail trade. In this example, we focus on the quarterly turnover of menswear shops during five quarters; see Table 7. The aggregates were determined before the influential suspect values were checked, which were detected using score functions. VAT turnover figures were eliminated if the editor thought that these were incorrect. It is difficult to correct VAT turnover figures, because these were not observed by Statistics Netherlands or according to CBS definitions, and contacting the reporters was not allowed.

Table 7. Estimated total quarterly turnover and turnover development of menswear shops in size class 10-40 for 1st quarter 2008 to 1st quarter 2009

Period	Total turnover (in mln euros)	Quarter-on-quarter development	Development compared to 1st quarter 2008
1st quarter 2008	120	-	-
2nd quarter 2008	154	29.1%	29.1%
3rd quarter 2008	136	-12.3%	13.2%
4th quarter 2008	174	28.3%	45.3%
1st quarter 2009	115	-33.6%	-3.6%

At first glance, the quarterly turnover figures seem plausible. Due to the sales in June and December, there is a relatively large amount of turnover in the second and fourth quarters. A quarterly turnover of 120 million is achieved if an adult man spends an average of 20 euros per quarter in a small to medium-sized menswear shop.

The quarter-on-quarter developments in Table 7 therefore fit in with the expected seasonal pattern for menswear shops. The same development is visible per size class. A negative year-on-year development for the 1st quarter in 2009 also fits in with the financial crisis situation. The developments therefore seem plausible. However, we should also look at influential suspect values, because the margin in which a development is plausible is quite large. The goal is to approximate the true development.

7.1.5 Quality indicators

We can calculate (7.1.1) and (7.1.4) before and after macro editing and determine the extent to which the relative deviation of an aggregate has decreased. The same applies for (7.1.3) and (7.1.5) if we are looking at ratios. If the relative deviation

remains comparable, then macro editing has had only little impact. However, it is possible that an aggregate/ratio approximates the actual situation, because a deviation compared to a reference aggregate/ratio can be correct. In this case, the macro editing may have led to the following:

- There is more confidence in the observed deviation at macro level;
- Improvements took place at micro level. This leads to better reference values for the editing of the following period.

Four percentages can be determined for one or more variables in the checked records:

- Percentage of records with an error detected;
- Percentage of records with an influential error detected;
- Percentage of records with a non-representative (correct) outlier detected;
- Percentage of records with a representative (correct) outlier detected.

7.2 Distribution method

7.2.1 Short description

In the distribution method, values of variables in a group of records are compared with one another using the univariate and multivariate distribution of these variables. This can be done with graphic tools or statistical measures. The outliers, the most suspect records, are then checked. If outliers turn out to be influential incorrect values, they are corrected. If an outlier is considered correct, then the question is whether it is representative for the population. In a sample of, for example, 1 in 10, an outlier is representative if nine comparable outliers occur in the population outside of the sample. If an outlier is not considered representative, then the weight is adapted. Methods for detecting outliers are discussed in Krieg and Smeets (2009).

7.2.2 Applicability

The primary goal is to detect influential errors which were missed or introduced during micro editing. The distribution method can be used for quantitative variables. If the distribution of the variables is not symmetric, then it is better to first transform the data, so that it better represents a normal distribution. Various distribution methods can otherwise create a biased impression.

7.2.3 Detailed description

For outlier detection, use can be made of various statistical measures (Project group Mesoanalyse, 2009). These provide insight into the distribution of the microdata and can be used to establish notable and/or structural changes in the microdata.

By themselves, statistical measures do not issue a judgement about the quality of the data. If, for example, there is a large distribution, this does not necessarily mean that the quality is poor. If the distribution in a publication cell is significantly larger compared to earlier periods, this can indeed indicate that the quality is not as high.

The measures below relate to a publication cell or a part thereof.

Representative value of a variable in the response:

- *Average*. This is often used, but is sensitive to outliers.
- *Moving average*. This is used, for example, the calculation of the "yardstick" for International Trade. The values for earlier (edited) periods are included in this.

Robust representative value in the response:

- *Truncated average*. If the value of a variable is smaller than *c* or larger than *d*, then eliminate the value. Now calculate the average. Truncation can be done one-sided (only a lower or upper limit) or two-sided.
- *Censored average*. If the value of a variable is smaller than *c*, then value = *c*. If the value of a variable is larger than *d*, then value = *d*. Now calculate the average. This can be done one-sided (only a lower or upper limit) or two-sided.
- *Median*. This is the value that lies at the midpoint of the data sorted based on the variable. This is extremely robust against outliers, especially if there are equal numbers of too low and too high values.

Measure for dispersion (of a variable) in the response:

- *Variance*. This measure is used very often, but is sensitive to outliers and does not have the same scale as the average.
- *Standard deviation (s.d.)*. This is also sensitive to outliers, but has the same scale as the average. This measure is equal to the root of the variance.
- *Range*. The difference between the minimum and maximum value.

Robust measure for dispersion in the response:

- *Interquartile distance*. The difference between the first quartile and third quartile of the cumulative distribution of a variable. This is used in box plots. This and the measure below are useful for symmetrical distributions.
- *(100 – α)% percentile minus the α% percentile*. $\alpha = 25$ indicates the interquartile distance. If there are many observations, then it is more convenient to use, for example, $\alpha = 5$.
- *Third quartile minus second quartile*. This is a dispersion measure for the values that are larger than the median. Just like the measure below, this is very useful in an asymmetrical distribution.
- *Second quartile minus first quartile*. This is a dispersion measure for the values that are smaller than the median (second quartile).

Measure for dispersion of the estimate of a population attribute:

- *Standard error of the estimator*. If this cannot be determined analytically, then it can be determined empirically using bootstrap or jackknife techniques.

Other distribution attributes:

- *Minimum*. The minimum value of a variable in the response. If, for example, this is negative and the variable can only be positive, then this means that part of the data is inconsistent.
- *Maximum*. The maximum value of a variable in the response. If, for example, this is extremely large, then this means that at least one value is suspect.
- *Skewness*. This indicates the extent to which the distribution of a variable is asymmetric. If the right tail area of the distribution is longer than the left tail area, then the skewness is positive. If the opposite is true, then the skewness is negative. Skewness can be caused by outliers.
- *Kurtosis*. This is high if the tail areas of the distribution are relatively long. Outliers lead to a high kurtosis.

Sufficient records must be available to reliably determine a distribution attribute. To accurately determine an interquartile distance, for example, more than 20 values are needed. A distribution attribute is particularly relevant if we combine it with other distribution attributes or compare the same attribute for different periods. The goal is to obtain a better understanding of the development in the publication figures.

It is interesting to compare each distribution attribute with the same attribute for an earlier period. If the attribute has changed considerably, then that is suspect. Relevant combinations of distribution attributes are:

- Range divided by interquartile distance
- Average divided by median
- Average divided by truncated/censored average
- Variance at t divided by variance at $t - 1$

If one of these is large, then there is at least one outlier.

Graphic tools, such as scatter and box plots, are also used to determine outliers. These Explorative Data Analysis (EDA) techniques are broadly applicable (Tukey, 1977) and available, for example, via Excel and SPSS. DesJardins (1997) illustrates applications of and useful additions to traditional EDA techniques.

There are also mathematical techniques to detect outliers such as the Mahalanobis distance; see Hoogland, Houbiers and De Waal (2002). Regression techniques can be used to obtain an impression of the relation between two variables. Standard regression techniques can produce a biased picture of the relation if there are outliers. In that case, it is better to use robust regression techniques. There are various robust regression techniques, such as M-estimators (Huber, 1981), the least median of squares method (Rousseeuw, 1984), the reweighted least squares method (Rousseeuw and Leroy, 1987) and generalised S-estimators (Croux, Rousseeuw and Hössjer, 1994). A number of these techniques are available in R, S-Plus, STATA and Matlab. M-estimators reduce the influence of outliers, but a single outlier may still be sufficient to disrupt these estimators. Several techniques are robust against outliers in the dependent variable, but not robust against outliers in the explanatory variables.

7.2.4 Example

At Statistics Netherlands, tax data is used increasingly often to compile business statistics. We are interested, for example, in the year-on-year development of VAT quarterly turnover figures. At record level, for the editing, we look at the VAT turnover and the growth rate; in other words, the VAT turnover in the reporting period divided by the VAT turnover in the reference period.

A histogram can be created to obtain an impression of the distribution of the VAT data; see Figure 4. Various attributes of the distribution can also be determined; see Table 8. The interquartile distance is a robust measure for the distribution. This means that this is not sensitive to outliers. Table 8 shows that at least one negative value occurs in size classes 21 and 30, and that the maximum turnover in size class 10 is conspicuously high. In addition, the VAT turnover has an asymmetrical distribution for size classes 10-30. This may be the result of outliers, certainly with skewness higher than 2.5 or smaller than –2.5. A kurtosis greater than 10 also indicates outliers.

Figure 4. Histogram of VAT quarterly turnover figures for 4th quarter 2008 for publication cell 47110; size class 22

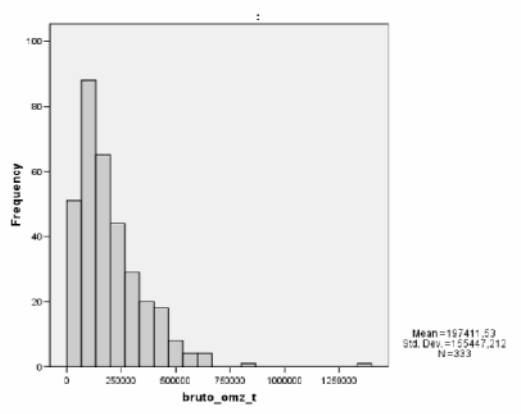


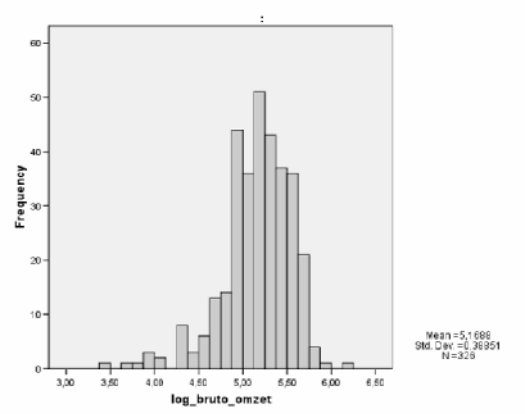
Table 8. Minimum, maximum, 1st, 2nd and 3rd quartile, interquartile distance (IQD), skewness and kurtosis of VAT turnover (in thousands of euros) of quarterly turnover figures of supermarkets in 4th quarter of 2008

[First row diagram below: SC / Min. / Max. / Quart 1 / Quart 2 / Quart 3 / IQD / Skewn. / Kurt.]

GK	Min.	Max.	Kwart 1	Kwart 2	Kwart 3	IKA	Scheefh.	Kurt.
10	0	2569	14	34	62	48	16,5	320,1
21	-177	1663	33	69	117	84	7,8	85,3
22	0	1349	91	163	273	182	2,0	9,1
30	-74	1873	212	331	492	280	2,1	6,9
40	0	2609	478	778	1343	865	0,6	-0,3
50	0	5245	1651	1960	2715	1064	0,5	1,3

If the data is very asymmetrically distributed, then it is advisable to first apply logarithmic transformation before performing graphic analyses. This will mainly apply for the growth rate. After this transformation, asymmetrically distributed data may be more symmetrically distributed, apart from outliers. In Figure 5, this can be seen for the VAT quarterly turnover of several supermarkets. We can now also observe that ten turnover figures in this size class are relatively small.

Figure 5. Histogram of logarithm of VAT quarterly turnover figures for 4th quarter 2008 for publication cell 47110; size class 22

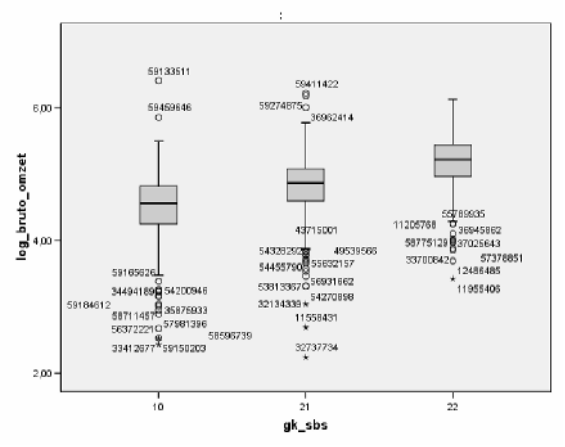


Suspect values can be detected using the diagrams listed below.

- Histogram of the values in a publication cell or stratum for year *t*.
- Scatter plot of the level or the growth rate for year *t* compared to the level or growth rate for year *t – 1*. In this case, we see only units for which a level or growth rate is available in the reporting period in the years *t* and *t – 1*.
- Two box plots (one-dimensional plot to detect outliers) next to each other: one with the level or growth rate for year *t* and one with the level or growth rate for year *t – 1*. Box plots for previous years may also be added.

Figure 6 shows an example made with SPSS with VAT turnover after a log transformation. This concerns box plots for supermarkets in three size classes. The '*' are extreme outliers, and the 'o' less extreme outliers. You can give each outlier a value (for example the enterprise id) so that you can find the record in the microdata. A large number of outliers on the lower half of a box plot indicates that the data is also not symmetrically distributed after a log transformation, and that we now have a distribution with a 'long left tail area'.

Figure 6. Box plots per size class of log10 (gross turnover) of supermarkets



8. References

Bankier, M., J.M. Fillion, M. Luc and C. Nadeau (1994), Imputing numeric and qualitative variables simultaneously. *Proceedings of the Section on Survey Research Methods*, American Statistical Association, 242-247.

Bankier, M. (2006), *Imputing numeric and qualitative variables simultaneously*. Memo, Statistics Canada, Social Survey Methods Division.

CANCEIS (2006), *CANCEIS version 4.5 user's guide*. Statistics Canada, Social Survey Methods Division.

Croux C., P. Rousseeuw and O. Hössjer (1994), Generalized S-estimators. *Journal of the American Statistical Association* 89, 1271-1281.

Daalmans, J. (2000), *Automatic error localisation of categorical data*. Research paper 0024, Statistics Netherlands, Voorburg.

DesJardins, D. (1997), *Experiences with introducing new graphical techniques for the analysis of census data*. UNECE Work Session on Statistical Data Editing, Prague.

Di Zio, M., U. Guarnera and O. Luzi (2005), *Improving the effectiveness of a probabilistic editing strategy for business data*. ISTAT, Rome.

Duin, C. van (2003), *Plausibiliteitsindicator voor Impect 2*. Internal report, Statistics Netherlands, Voorburg.

Farwell, K. (2005), *Significance editing for a variety of survey situations*. Paper presented at the 55th session of the International Statistical Institute, Sydney.

Fellegi, I.P. and D. Holt (1976), A systematic approach to automatic edit and imputation. *Journal of the American Statistical Association* 71, 17-35.

Granquist, L. and J. G. Kovar (1997), Editing of Survey Data: How Much Is Enough? In L. Lyberg, P. Biemer, M. Collins, E. de Leeuw, C. Dippo, N. Schwarz and D. Trewin (eds.), *Survey Measurement and Process Quality*. New York: Wiley, pp. 415-435.

Haar, M. ter (2002), *IMPECT 2: automatische correctie, version 0.2 (rev. 1)*. Internal report, Statistics Netherlands, Heerlen.

Hedlin, D. (2003), Score Functions to Reduce Business Survey Editing at the U.K. Office for National Statistics. *Journal of Official Statistics* 19, 177-199.

Hedlin, D. (2008), *Local and global score functions in selective editing*. UNECE Work Session on Statistical Data Editing, Vienna, working paper no. 31.

Hidiroglou, M. A. and J.M. Berthelot (1986), Statistical Editing and Imputation for Periodic Business Surveys. *Survey Methodology* 12 (1), 73-83.

Hoogland, J. (2002), *Selective editing by means of Plausibility Indicators*. UNECE Work Session on Statistical Data Editing, Helsinki, working paper no. 33.

Hoogland, J., M. Houbiers and T. de Waal (2002), *Syllabus bij de cursus gaafmaakmethoden en software voor bedrijfseconomische statistieken. Version 2*. Internal report, Statistics Netherlands, Voorburg.

Hoogland, J. and R. Smit (2008), *Selective automatic editing of mixed mode questionnaires for structural business statistics*. UNECE Work Session on Statistical Data Editing, Vienna, working paper no. 2.

Huber, P. (1981), *Robust statistics*. Wiley, New York.

ISTAT, CBS and SFSO (2007), *Recommended Practices for Editing and Imputation in Cross-Sectional Business Surveys* (http://edimbus.istat.it/EDIMBUS1/document/RPM_EDIMBUS).

Jong, A. de (2002), *UNI-EDIT: Standardized processing of structural business statistics in the Netherlands*. UNECE Work Session on Statistical Data Editing, Helsinki, working paper no. 27.

Krieg, S. and M. Smeets (2009), *Representative outliers*. Methods Series document, Statistics Netherlands, Heerlen [English translation from Dutch in 2011].

Latouche, M. and J.M. Berthelot (1992), Use of a score function to prioritise and limit recontacts in editing business surveys. *Journal of Official Statistics* 8, 389-400.

Lawrence, D. and R. McKenzie (2000), The general application of significance editing. *Journal of Official Statistics* 16, 243-253.

Loo, M. van der and J. Pannekoek (2007), *Advies gaafmaken en imputeren van de statistiek Bouwobjecten in Voorbereiding*. Internal report, Statistics Netherlands, Voorburg.

Loo, M. P. J. van der (2008), *An analysis of editing strategies for mixed-mode establishment surveys*. Discussion paper (08004), Statistics Netherlands, Voorburg.

Pannekoek, J. and C. Tempelman (2005), *Evaluatie van imputatiemethoden voor IMPECT: deductieve imputatie en correctie voor overduidelijke fouten*. Internal report, Statistics Netherlands, Voorburg.

Pannekoek, J., C. Harmsen, M. van Huis and K. Prins (2008), *Automatisch gaafmaken van GBA-gegevens met de "Nearest-neighbor Imputation Methodology"*. Internal report, Statistics Netherlands, The Hague.

Pol, F. van de, F. Bakker and T. de Waal (1997), *On principles for automatic editing of numerical data with equality checks*. Statistics Netherlands, Voorburg.

Projectgroep Mesoanalyse (2009), *HEcS+ Mesoanalyse: Analyse en interactief gaafmaken in de HEcS-keten Versie 0.1p3*. Internal report, Statistics Netherlands.

Rousseeuw, P. (1984), Least median of squares regression. *Journal of the American Statistical Association* 79, 871-880.

Rousseeuw, P. and A. Leroy (1987), *Robust regression and outlier detection*. Wiley series in probability and mathematical statistics.

Scholtus, S. (2007), *Automatische correctie van tekenfouten en verwisselingen van baten en lasten*. Internal report, Statistics Netherlands, Voorburg.

Scholtus, S. (2008a), *Algorithms for correcting some obvious inconsistencies and rounding errors in business survey data*. Discussion paper (08015), Statistics Netherlands, The Hague.

Scholtus, S. (2008b), *Automatisch gaafmaken van GBA-gegevens met CANCEIS*. Internal report, Statistics Netherlands, The Hague.

Scholtus, S. (2009), *Automatic correction of simple typing errors in numerical data with balance edits*. Discussion paper (09046), Statistics Netherlands, The Hague.

Sluis, W. (2004), *SLICEDemo*. Internal report, Statistics Netherlands, Voorburg.

Stoop, J.R. (2003), *Het lekkerste stuk uit CherryPie. Selectie van een optimale oplossing bij automatisch gaafmaken*. Internal report, Statistics Netherlands, Voorburg.

Tukey, J. (1977), *EDA: Exploratory Data Analysis*. Addison-Wesley, Massachusetts.

Waal, T. de and R. Quere (2003), A fast and simple algorithm for automatic editing for mixed data. *Journal of Official Statistics* 19, 383-402.

Waal, T. de (2003), *Processing of erroneous and unsafe data*. Doctoral thesis, Erasmus University Rotterdam.

Waal, T. de (2005a), *SLICE 1.5: Software voor automatisch gaafmaken en imputeren*. Internal report, Statistics Netherlands, Voorburg.

Waal, T. de (2005b), *De methodologie van SLICE 1.5: Het algoritme van Cherry Pie*. Internal report, Statistics Netherlands, Voorburg.

Waal, T. de (2008), *An overview of statistical data editing*. Discussion paper (08018), Statistics Netherlands, The Hague.

Version history

Version	Date	Description	Authors	Reviewers
Dutch version: Controle en correctie				
1.0	25-01-2010	First Dutch version	Jeffrey Hoogland Mark van der Loo Jeroen Pannekoek Sander Scholtus	Guus van de Burgt Micha Juffermans Roos Smit
English version: Data editing: detection and correction of errors				
1.0E	17-02-2011	First English version	Jeffrey Hoogland Mark van der Loo Jeroen Pannekoek Sander Scholtus	