WP.42
ENGLISH ONLY

UNITED NATIONS
ECONOMIC COMMISSION FOR EUROPE

CONFERENCE OF EUROPEAN STATISTICIANS

Work Session on Statistical Data Editing
(Ljubljana, Slovenia, 9-11 May 2011)

Topic (vii): New and emerging methods

Automatic Editing with Soft Edits

Invited Paper

Prepared by Sander Scholtus, Statistics Netherlands

I. Introduction

1. The goal of automatic editing is to accurately detect and correct errors and missing values in a data file in a fully automated manner, i.e. without human intervention. Provided that automatic editing leads to data of sufficient quality, it can be used as a partial alternative to manual editing, and thereby be an important tool for increasing the efficiency of a statistical process. In practice, automatic editing implies that the data is made consistent with respect to a set of constraints (edits). Examples of edits are:

$$\textit{Profit} = \textit{Total Turnover} - \textit{Total Costs}, \tag{1}$$

and

$$\textit{Profit} \leq 0.6 \times \textit{Total Turnover}. \tag{2}$$

Note that there is a conceptual difference between these two edits. Edit (1) is an example of an edit that has to hold by definition, so that every combination of values that fails this edit necessarily contains an error. Such edits are commonly known as hard edits, fatal edits, or logical edits. Edit (2) is an example of an edit that identifies combinations of values that are implausible, but not necessarily incorrect. In this example, records for which *Profit* is larger than 60% of *Total Turnover* are considered suspicious. However, it is conceivable that such a combination of values is occasionally correct. Edits of this type, which do not identify errors with certainty, are known as soft edits, or query edits.

2. Current algorithms for automatic editing used by NSIs – including the error localisation module of SLICE at Statistics Netherlands – are often based on the well-known Fellegi-Holt paradigm. A limitation of these algorithms is that they necessarily treat all edits as hard edits. That is to say, a failed edit is always attributed to an error in the data. In manual editing, however, subject-matter specialists also make extensive use of soft edits. During automatic editing, these soft edits are either not used at all, or else interpreted as hard edits. Both solutions are unsatisfactory, because in the first case some errors are missed during automatic editing, and in the second case some correct values are wrongfully identified as erroneous. In fact, the inability of automatic editing methods to handle soft edits partly explains why many differences between manually edited and automatically edited data are found in practice.

3. The object of this paper is to present a new formulation of the automatic error localisation problem, which can distinguish between hard edits and soft edits. In addition, it is shown how the error localisation algorithm of SLICE can be adapted to solve this new error localisation problem. The remainder of this paper is organised as follows. Section II provides a brief summary of methods for solving the error localisation problem based on the Fellegi-Holt paradigm. In Section III, a distinction between hard and soft edits is introduced and the theory is extended to this situation, leading to a new error localisation algorithm. In Section IV, the new algorithm is illustrated by means of a small example. A brief discussion in Section V concludes the paper.

II. Background: the Error Localisation Problem and How to Solve It

4. For the sake of brevity, it is assumed throughout this paper that the data consists of real-valued numerical variables $(x_1,\ldots,x_p)$, and that each edit ψ^k can be written as a linear inequality:

$$\psi^k: \quad a_{k1}x_1 + \cdots + a_{kp}x_p + b_k \geq 0. \tag{3}$$

Suppose that a record $(x_1^0,\ldots,x_p^0)$ of unedited data is given, possibly containing both errors and missing values. It is straightforward to determine which edits are failed and which are satisfied. Given that at least one edit is failed, the problem of finding the erroneous values that are causing the edit failures (i.e. the error localisation problem) is much more difficult.

5. In order to solve the error localisation problem automatically, a formal strategy for finding the erroneous variables has to be adopted. According to the famous (generalised) Fellegi-Holt paradigm (Fellegi and Holt, 1976), one should find a subset of the variables which (a) can be imputed such that the adjusted record $(x_1,\ldots,x_p)$ satisfies all edits, and (b) minimises the following target function:

$$D_{FH} = \sum_{j=1}^{p} w_j I(x_j \neq x_j^0), \tag{4}$$

where w_j denotes the confidence weight of variable x_j, and $I(.)$ equals 1 if its argument is true, and 0 otherwise. The confidence weights are an a priori measure of trust in the unedited values of different variables: a variable with a high confidence weight supposedly contains relatively few errors, a variable with a low weight supposedly contains relatively many. The confidence weights are usually chosen by subject-matter experts. The original Fellegi-Holt paradigm is recovered by taking all confidence weights equal (for instance equal to one), so that the number of imputed variables is minimised.

6. Clearly, a subset of variables can only be a solution to the error localisation problem if every failed edit involves at least one of these variables, i.e. if the failed edits are 'covered' by the subset of variables. However, this condition is not sufficient. Fellegi and Holt (1976) showed that, in order to determine whether a combination of variables can be imputed to satisfy all edits, it is necessary to derive so-called essentially new implied edits from the original set of edits.

7. The importance of these implied edits can be seen in the following small example. Suppose that a record of three variables has to satisfy the two inequality edits $x_1 \geq x_2$ and $x_2 \geq x_3$. The unedited record $(x_1^0, x_2^0, x_3^0) = (4,8,5)$ fails the first edit and satisfies the second edit. Since x_2 is involved in the failed edit, one might try to solve the edit failure by changing this variable. However, this is impossible without causing the second edit to become failed, because the imputed value has to satisfy $x_2 \leq 4$ and $x_2 \geq 5$. There is in fact an essentially new implied edit that can be derived from the two original edits: $x_1 \geq x_3$. The implied edit is failed by the original record and x_2 is not involved in this edit. This shows that only changing the value of x_2 is not a solution to the error localisation problem for this record.

8. For numerical edits of the form (3), essentially new implied edits are derived by applying a technique called Fourier-Motzkin elimination, which tries to eliminate a chosen variable from all pairs of edits that involve this variable. Suppose that x_g is involved in ψ^s and ψ^t, and suppose that $a_{sg}a_{tg} < 0$, i.e. the coefficients of x_g in the two edits have opposite signs. If it does not hold that $a_{sg}a_{tg} < 0$, then x_g cannot be eliminated from ψ^s and ψ^t. It can be assumed without loss of generality that $a_{sg} < 0$ and $a_{tg} > 0$. This means that ψ^s can be written as an upper bound on the value of x_g, given the values of the other variables:

$$x_g \leq \frac{1}{-a_{sg}}(a_{s1}x_1 + \cdots + a_{s,g-1}x_{g-1} + a_{s,g+1}x_{g+1} + \cdots + a_{sp}x_p + b_s). \tag{5}$$

In the same way, ψ^t can be written as a lower bound on x_g:

$$x_g \geq \frac{1}{-a_{tg}}(a_{t1}x_1 + \cdots + a_{t,g-1}x_{g-1} + a_{t,g+1}x_{g+1} + \cdots + a_{tp}x_p + b_t). \quad (6)$$

Combining the two bounds, in the spirit of the example from the previous paragraph, leads to a new edit:

$$\frac{1}{-a_{sg}}(a_{s1}x_1 + \cdots + a_{s,g-1}x_{g-1} + a_{s,g+1}x_{g+1} + \cdots + a_{sp}x_p + b_s)$$

$$\geq \frac{1}{-a_{tg}}(a_{t1}x_1 + \cdots + a_{t,g-1}x_{g-1} + a_{t,g+1}x_{g+1} + \cdots + a_{tp}x_p + b_t)$$

which can be written in the general form (3) as

$$\psi^*: \quad a_1^*x_1 + \cdots + a_p^*x_p + b^* \geq 0, \quad (7)$$

with $a_j^* = a_{tg}a_{sj} - a_{sg}a_{tj}$ and $b^* = a_{tg}b_s - a_{sg}b_t$. Note that $a_g^* = 0$, so x_g is not involved in the implied edit.

9. In the original method of Fellegi and Holt, the above elimination procedure is repeatedly applied to the original set of edits until no more essentially new implied edits can be derived. Together, the original edits and the essentially new edits form the so-called complete set of edits. Fellegi and Holt (1976) prove that a subset of variables that covers all failed edits from the complete set is a feasible solution to the error localisation problem. Thus, once a complete set of edits has been obtained, the error localisation problem can be solved in a straightforward manner, for any record, by finding the subset of variables that minimises D_{FH} among all subsets that cover the failed edits in the complete set of edits. However, the complete set of edits can be extremely large in practice, which means that the original method of Fellegi and Holt is not always computationally feasible.

10. De Waal and Quere (2003) describe a different, but related error localisation method, which makes use of implied edits without deriving the complete set of edits. A branch-and-bound algorithm is run for each record separately. The algorithm can be represented as a binary tree, where a different set of edits is associated with each node. At the root node of the tree, the current set of edits is the original set of edits, say Ψ_0. In each node, a variable is selected that has not been selected in any of its predecessor nodes, and two branches are constructed. In the first branch, it is assumed that the original value of the selected variable is erroneous, and the variable is eliminated from the current set of edits, say Ψ_q, by means of Fourier-Motzkin elimination. This generates a new set of edits Ψ_{q+1}, which consists of the essentially new implied edits obtained by eliminating the selected variable and the edits from Ψ_q that do not involve the selected variable. In the second branch, it is assumed that the original value of the selected variable is correct. Here, a new set of edits Ψ_{q+1} is generated from Ψ_q by fixing the selected variable to its original value in all edits that involve this variable. In both cases the variable has been treated, and it is not involved in the current set of edits anymore.

11. Once all variables have been treated (i.e. either eliminated or fixed), the algorithm reaches a terminal node of the tree. The associated set of edits in a terminal node does not contain any variables, hence it must either be empty or consist of elementary relations such as $1 \geq 0$ and $0 \geq 1$. As the latter example shows, some of these elementary relations may be self-contradicting. De Waal and Quere (2003) show that the feasible solutions to the error localisation problem correspond with terminal nodes that contain no self-contradicting relations. If a terminal node contains no self-contradicting relations, then the variables that have been eliminated to reach this node can be imputed to satisfy the original set of edits. This property follows by a repeated application of the following theorem.

Theorem 1. *Consider a node in the binary tree with an associated set of edits Ψ_q, and let T_q be the index set of variables that have not been treated yet. Suppose that x_g is either eliminated or fixed to obtain the next node, with the associated set of edits Ψ_{q+1}, and define $T_{q+1} := T_q \setminus \{g\}$. Then there exist values u_j for the variables with $j \in T_{q+1}$ that satisfy all edits in Ψ_{q+1}, if and only if there also exists a value u_g such that the values u_j for $j \in T_q$ satisfy all edits in Ψ_q.*

See De Waal (2003, pp. 133-135) or De Waal et al. (2011, pp. 135-138) for a proof.

12. Since the number of terminal nodes in the binary tree increases exponentially with the number of variables in the data, it is important to reduce the amount of computational work as much as possible by pruning the tree. Branches can be pruned as soon as it becomes clear that they either do not lead to feasible solutions, or only to solutions for which the value of D_{FH} is higher than the best solution found so far. Another way to reduce the size of the tree is to start by eliminating the variables with missing values, since these cannot be fixed to their original value. Variables with missing values certainly have to be imputed, so they occur in every feasible solution to the error localisation problem.

13. The branch-and-bound algorithm of De Waal and Quere (2003) has been implemented in the software package SLICE at Statistics Netherlands. It is currently used for the automatic editing of the Dutch structural business statistics. Other examples of software packages for automatic editing based on the Fellegi-Holt paradigm are: GEIS and its successor Banff (Banff Support Team, 2003), SPEER (Winkler and Draper, 1997), and AGGIES (Todaro, 1999). While the algorithms of these software packages differ in some respects, they all try to solve the error localisation problem by finding a minimal set of variables that can be imputed to satisfy all specified edits simultaneously. This means that they necessarily treat all edits as hard edits.

III. Using Soft Edits in Automatic Error Localisation

A. A Short Theory of Edit Failures

14. A fundamental property of the Fourier-Motzkin elimination technique is the fact that the values of $x_1,\ldots,x_{g-1},x_{g+1},\ldots,x_p$ satisfy the implied edit (7), if and only if there exists a value for x_g which, together with the values of the other variables, satisfies both edits (5) and (6). In fact, this feature forms the basis for the proof of Theorem 1. Looking at this equivalence from another point of view, if the values of $x_1,\ldots,x_{g-1},x_{g+1},\ldots,x_p$ do *not* satisfy the implied edit (7), then it holds that

$$\frac{1}{-a_{sg}}(a_{s1}x_1+\cdots+a_{s,g-1}x_{g-1}+a_{s,g+1}x_{g+1}+\cdots+a_{sp}x_p+b_s)$$
$$<\frac{1}{-a_{tg}}(a_{t1}x_1+\cdots+a_{t,g-1}x_{g-1}+a_{t,g+1}x_{g+1}+\cdots+a_{tp}x_p+b_t)$$

and hence it is impossible to satisfy edits (5) and (6) simultaneously. However, it is still possible in this case to find a value for x_g that satisfies either edit (5) or edit (6). While this observation in itself is almost trivial, it forms the basis for the proof of Theorem 2 below.

15. Suppose that, at some point during an execution of the branch-and-bound algorithm of De Waal and Quere (2003), q variables have been treated (i.e. either eliminated from the original edits or fixed). The current set of edits is denoted by Ψ_q, and the edits in this set are denoted by ψ_q^k. It is possible to associate with each current edit ψ_q^k a set B_q^k, which contains the indices of all the original edits that were used, either directly or indirectly, to derive this edit. In fact, B_q^k is defined recursively as follows:

- For an original edit ψ_0^k, define $B_0^k := \{k\}$.
- For an edit ψ_q^k that is derived from another edit ψ_{q-1}^l either by fixing a variable to its original value or by simply copying the edit, define $B_q^k := B_{q-1}^l$.
- For an edit ψ_q^k that is derived by eliminating a variable from two other edits ψ_{q-1}^s and ψ_{q-1}^t, define $B_q^k := B_{q-1}^s \cup B_{q-1}^t$.

16. A set B is called a representing set of a collection of index sets $B_q^{k_1},\ldots,B_q^{k_r}$, if it contains at least one element from each of $B_q^{k_1},\ldots,B_q^{k_r}$ (see, e.g., Mirsky, 1971, p. 25). Note that in this case the elements of B refer to a subset of the original edits. The following theorem can now be proved.

Theorem 2. *Suppose that the current set of edits can be partitioned as* $\Psi_q = \Psi_q^{(1)} \cup \Psi_q^{(2)}$, *where the edits from* $\Psi_q^{(1)}$ *are satisfied by the original values of the variables that have not been treated, and the edits in* $\Psi_q^{(2)}$ *are failed by these values. Also, suppose that* B *is a representing set of the index sets* B_q^k *for all* $\psi_q^k \in \Psi_q^{(2)}$. *Then there exist values for the eliminated variables which, together with the original values of the other variables, satisfy all the original edits except those in* B.

The proof of Theorem 2 is given in the appendix to this paper, because it is somewhat technical.

17. The importance of this theorem is that it enables one to evaluate, at each stage of the branch-and-bound algorithm, which combinations of the original edits could be satisfied by imputing the variables that have been eliminated so far, and also which edits would remain failed. If it is possible to impute the eliminated variables such that only soft edits remain failed, but the hard edits are all satisfied, then imputing these variables can be considered as a feasible solution to the error localisation problem. This idea will be elaborated in the next subsection.

B. A New Error Localisation Problem

18. It is now assumed that the original set of edits Ψ_0 has been partitioned into two disjoint subsets: $\Psi_0 = \Psi_{0H} \cup \Psi_{0S}$. The edits $\psi_{0H}^k \in \Psi_{0H}$ are hard edits, the edits $\psi_{0S}^k \in \Psi_{0S}$ are soft edits. A subset of the variables is considered as a feasible solution to the error localisation problem if it can be imputed to satisfy all the edits in Ψ_{0H}. Since the theory from the previous subsection can be used to determine which edits from Ψ_{0S} remain failed (if any), it is possible to use information about the soft edit failures in choosing the optimal solution to the error localisation problem. This can be done by adding a second term to target function (4):

$$D = D_{FH} + D_{soft}, \tag{8}$$

where D_{soft} represents the costs that are associated with failed soft edits.

19. Instead of minimising the (weighted) number of imputed variables that must be imputed to satisfy all the edits, the objective now becomes to find the subset of variables that minimises D among those subsets that can be imputed to satisfy all hard edits. Depending on the choice of the confidence weights and of D_{soft}, it may happen that the optimal solution to the new error localisation problem imputes more variables than is strictly needed to satisfy all hard edits, provided that this leads to a smaller value of D, because some of the failed soft edits also become satisfied.

20. Probably the easiest way to define the costs of soft edit failures, is to associate a fixed failure weight s_k with each soft edit, and to define D_{soft} as the sum of the failure weights of the soft edits that remain failed:

$$D_{soft} = \sum_{k=1}^{K_S} s_k I(\psi_{0S}^k \text{ is failed}), \tag{9}$$

where K_S denotes the number of soft edits that have been specified. The failure weights can be chosen by subject-matter experts, analogously to the confidence weights in the generalised Fellegi-Holt paradigm. That is to say, the failure weight expresses the importance that is attached to a soft edit from a subject-matter related point of view.

21. A drawback of using fixed failure weights is that they do not take the size of the edit failures into account: every record that fails a particular soft edit ψ_{0S}^k receives the same contribution to D_{soft}, namely

s_k. This differs from the way soft edits are interpreted by human editors during interactive editing. There, a failed soft edit points to a combination of values that is suspicious, and the degree of suspicion depends heavily on the size of the edit failure: a small failure is ignored more easily than a large failure. This point will be taken up in Section V.

C. Solving the New Error Localisation Problem

22. The new error localisation problem, with a distinction between hard and soft edits, can be solved by a modified version of the branch-and-bound algorithm of De Waal and Quere (2003). In the root node of the binary tree, the original edits are partitioned into a set of hard edits, Ψ_{0H}, and a set of soft edits, Ψ_{0S}. To each soft edit $\psi_{0S}^k \in \Psi_{0S}$ there is associated an index set $B_{0S}^k := \{k\}$. The number of treated variables is initially $q := 0$.

23. If the current node of the binary tree is not a terminal node, then an untreated variable is selected, say x_g. Two new branches are generated. In the first branch, x_g is eliminated from the current set of edits $\Psi_q = \Psi_{qH} \cup \Psi_{qS}$ by means of Fourier-Motzkin elimination, and in the second branch, x_g is fixed to its original value. Both procedures are carried out largely the same way as in the original algorithm. The main difference is that the new algorithm distinguishes between hard and soft implied edits.

24. In the branch where x_g is fixed to its original value, a new set of edits $\Psi_{q+1} = \Psi_{q+1,H} \cup \Psi_{q+1,S}$ is obtained, where $\Psi_{q+1,H}$ is derived from Ψ_{qH}, and $\Psi_{q+1,S}$ is derived from Ψ_{qS}. In addition, define $B_{q+1,S}^k := B_{qS}^l$ for an edit $\psi_{q+1,S}^k$ that is generated from ψ_{qS}^l in this way.

25. In the branch where x_g is eliminated from the edits, the new set of hard edits $\Psi_{q+1,H}$ consists of all edits from Ψ_{qH} that do not involve x_g, plus all implied edits that are obtained by eliminating x_g from a combination of only hard edits. The new set of soft edits $\Psi_{q+1,S}$ contains all the other edits, i.e.

- all edits $\psi_{qS}^l \in \Psi_{qS}$ that do not involve x_g: define $B_{q+1,S}^k := B_{qS}^l$ for an edit $\psi_{q+1,S}^k$ that is generated in this way;
- all implied edits that are obtained from a combination of only soft edits, say ψ_{qS}^s and ψ_{qS}^t: define $B_{q+1,S}^k := B_{qS}^s \cup B_{qS}^t$ for an edit $\psi_{q+1,S}^k$ that is generated in this way;
- all implied edits that are obtained from a combination of a soft edit and a hard edit, say ψ_{qS}^s and ψ_{qH}^t: define $B_{q+1,S}^k := B_{qS}^s$ for an edit $\psi_{q+1,S}^k$ that is generated in this way.

26. After generating the new set of edits Ψ_{q+1}, it is checked whether any of these edits are failed by the original values of the variables that have not been treated yet. In the new algorithm, three possible situations may arise here. First of all, if at least one edit in $\Psi_{q+1,H}$ remains failed, then the variables that have been eliminated so far cannot be imputed to satisfy all the original hard edits. In this case, define $q := q+1$ and continue the generation of branches from the current node.

27. A second possibility is that none of the edits in Ψ_{q+1} remain failed. This means that the variables that have been eliminated so far can be imputed to satisfy all the original edits, both hard and soft. This means that a feasible solution to the error localisation problem has been found. The value of the target function equals $D = D_{FH}$, i.e. the sum of the confidence weights of the eliminated variables. If this value is smaller than (or equal to) the value $D_{\min}$ of the best solution found so far, then the new solution is kept. Otherwise, it is discarded. Either way, it is not useful to continue the algorithm from the current node, because the value of the target function can only increase if more variables are eliminated.

28. The third and final possibility is that the edits in $\Psi_{q+1,H}$ are satisfied, but at least one of the edits in $\Psi_{q+1,S}$ remains failed. In this case, the variables that have been eliminated so far can be imputed to satisfy the original hard edits, but not all the original soft edits. Hence a feasible solution to the error localisation problem has been found, but the contribution of D_{soft} to the target function is non-zero.

29. According to Theorem 2, it is possible to satisfy all original soft edits except those in a representing set B of the sets $B^k_{q+1,S}$ for the failed edits from $\Psi_{q+1,S}$. Since this property is shared by all representing sets, it is possible to choose B in such a way that D_{soft} is minimised. If expression (9) is used for D_{soft}, then the optimal choice of B can be found by solving a minimisation problem:

$$\begin{aligned} &\min \sum_{k=1}^{K_S} s_k z_k, \text{ such that} \\ &\sum_{k \in B^l_{q+1,S}} z_k \geq 1, \text{ for all violated } \psi^l_{q+1,S} \in \Psi_{q+1,S} \\ &z_k \in \{0,1\}, \text{ for all } k = 1,\ldots,K_S \end{aligned} \tag{10}$$

This problem can be solved by applying a (much simpler) branch-and-bound algorithm to explore all possible choices of $z_1,\ldots,z_{K_S}$. The associated minimal representing set is $B = \{k \mid z_k = 1\}$. Moreover, the contribution of D_{soft} to D equals the minimal value of problem (10).

30. Like in the previous case, the value of D is compared to that of the best solution found so far. If $D > D_{\min}$, then the current solution is discarded. Regardless of this, however, it is meaningful to continue the algorithm from the current node, because eliminating more variables may lead to a solution with a lower value of D: an increase in D_{FH} may be compensated by a decrease in D_{soft}. Therefore, define $q := q+1$ and continue the generation of branches from the current node.

31. The correctness of this branch-and-bound algorithm follows from the theory of Section III.A. Note that the index sets B^k_q from Theorem 2 only have to be calculated for the soft edits, because a subset of the variables is never considered as a feasible solution to the error localisation problem when at least one of the hard edits remains failed. This means that, in every application of Theorem 2, all implied edits that are derived from hard edits must be contained in $\Psi^{(1)}_q$. Also note that the new algorithm reduces to the original algorithm of De Waal and Quere (2003) if no soft edits have been specified.

IV. Example

32. To illustrate the new branch-and-bound algorithm, it is applied to a very small example. In this example, which is based on an example from De Waal (2003), there are four numerical variables: total turnover (T), profit (P), total costs (C), and total number of employees (N). These variables have to satisfy six hard edits and two soft edits:

$$\begin{aligned} \psi^1_{0H}: &\quad T - C - P \geq 0 \\ \psi^2_{0H}: &\quad -T + C + P \geq 0 \\ \psi^3_{0H}: &\quad T \geq 0 \\ \psi^4_{0H}: &\quad C \geq 0 \\ \psi^5_{0H}: &\quad N \geq 0 \\ \psi^6_{0H}: &\quad 550N - T \geq 0 \\ \psi^1_{0S}: &\quad 0.5T - P \geq 0 \qquad (B^1_{0S} = \{1\}) \\ \psi^2_{0S}: &\quad P + 0.1T \geq 0 \qquad (B^2_{0S} = \{2\}) \end{aligned}$$

Note that the first two hard edits together are equivalent to $T - C = P$. The unedited record $(T^0, P^0, C^0, N^0) = (100,40000,60000,5)$ is inconsistent, because it fails the first hard edit and the first soft

edit. The confidence weights of the variables are $(w_T, w_P, w_C, w_N) = (2,1,1,3)$, and the failure weights of the two soft edits are $s_1 = s_2 = 2$.

33. Suppose that the variable P is selected first. In the branch where P is eliminated from the original edits, the following set of new edits is obtained:

$\psi_{1H}^1: \quad T \geq 0$
$\psi_{1H}^2: \quad C \geq 0$
$\psi_{1H}^3: \quad N \geq 0$
$\psi_{1H}^4: \quad 550N - T \geq 0$
$\psi_{1S}^1: \quad -0.5T + C \geq 0 \qquad (B_{1S}^1 = \{1\})$
$\psi_{1S}^2: \quad 1.1T - C \geq 0 \qquad (B_{1S}^2 = \{2\})$
$\psi_{1S}^3: \quad 0.6T \geq 0 \qquad (B_{1S}^3 = \{1,2\})$

For instance, ψ_{1S}^2 is obtained by eliminating P from ψ_{0H}^1 and ψ_{0S}^2. The last soft edit ψ_{1S}^3 is in fact equivalent to ψ_{1H}^1, which means that it can be discarded.

34. If the original values $(T^0, C^0, N^0) = (100,60000,5)$ are used to evaluate the current set of edits, it is seen that the hard edits are all satisfied. This shows that identifying only the original value of P as erroneous is a feasible solution to the error localisation problem. However, ψ_{1S}^2 remains failed. Since $B = \{2\}$ is a representing set of B_{1S}^2, Theorem 2 states that it is possible to impute a value for P which satisfies all the original edits except for ψ_{0S}^2. This is in fact the minimal representing set according to problem (10). The value of target function (8) therefore equals $D = D_{FH} + D_{soft} = w_P + s_2 = 1 + 2 = 3$.

35. Possibly, the current solution can be improved by eliminating another variable, say C, from the current set of edits. This yields:

$\psi_{2H}^1: \quad T \geq 0$
$\psi_{2H}^2: \quad N \geq 0$
$\psi_{2H}^3: \quad 550N - T \geq 0$
$\psi_{2S}^1: \quad 1.1T \geq 0 \qquad (B_{2S}^1 = \{2\})$
$\psi_{2S}^2: \quad 0.6T \geq 0 \qquad (B_{2S}^2 = \{1,2\})$

The two new soft edits are both redundant, because they are equivalent to hard edit ψ_{2H}^1. In fact, the original values $(T^0, N^0) = (100,5)$ satisfy all the new edits. This means that the variables P and C can be imputed to satisfy all the original edits, including the soft edits. The value of target function (8) for this solution to the error localisation problem equals $D = D_{FH} = w_P + w_C = 1 + 1 = 2$. This is an improvement compared to the previous solution. Moreover, this solution cannot be improved further by eliminating more variables in the current branch of the binary tree.

36. So far, one branch of the binary tree has been considered. If the algorithm is continued by exploring all the other branches, it turns out that the best solution found so far (impute P and C) is also the optimal solution. A possible way to impute the record is: $(T, P, C, N) = (100,40,60,5)$. This solution has the nice interpretation that the original values of P and C were overstated by a factor of 1,000.

37. It is of interest to note that, if the two soft edits are not used in this example, then the first solution found above, i.e. impute only P, is the optimal solution to the error localisation problem for this record. In this case, the record has to be imputed as: $(T, P, C, N) = (100,-59900,60000,5)$. This illustrates that, in this example at least, the soft edits are important for finding imputations that are not just consistent with the hard edits, but also plausible.

V. Discussion

38. In this paper, a new formulation of the error localisation problem has been proposed, which takes the distinction between hard edits and soft edits into account. Also, it was shown how the branch-and-bound algorithm of De Waal and Quere (2003) can be modified to solve the new error localisation problem. For the sake of brevity, the description in this paper was limited to numerical variables with edits in the form of linear inequalities. This restriction is not necessary, however: just like the original algorithm of De Waal and Quere (2003), the new algorithm can also be applied to categorical and mixed data. The interested reader is referred to Scholtus (2011) for details.

39. The above description shows that it is theoretically possible to take the distinction between hard and soft edits into account in automatic editing. It remains to be seen whether the new error localisation algorithm is also computationally feasible in practice. A prototype implementation of the new algorithm is currently being made, using the R programming language, in order to answer this question. Assuming that the algorithm is computationally feasible, tests with realistic data also have to be conducted to see whether the new approach can be used to improve the quality of automatically edited data. Possibly, some refinements are necessary to make the new methodology work in practice.

40. It is still an open problem how the costs of soft edit failures can best be modelled, i.e. how the term D_{soft} in (8) should be defined. As mentioned above, an approach that takes the size of the edit failures into account is intuitively more appealing than using fixed failure weights. Scholtus (2011) shows how the size of the edit failures can be taken into account through dynamic failure weights, which are updated during an execution of the algorithm. Unfortunately, this makes the algorithm more complex.

VI. Literature

Banff Support Team (2003), Functional Description of the Banff System for Edit and Imputation. Technical report, Statistics Canada.

De Waal, T. (2003), Processing of Erroneous and Unsafe Data. PhD Thesis, Erasmus University, Rotterdam.

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

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

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

Mirsky, L. (1971), *Transversal Theory*. Academic Press, Inc., New York.

Scholtus, S. (2011), Automatic Editing with Soft Edits. Discussion Paper (forthcoming), Statistics Netherlands, The Hague.

Todaro, T.A. (1999), Overview and Evaluation of the AGGIES Automated Edit and Imputation System. Working Paper, UN/ECE Work Session on Statistical Data Editing, Rome, Italy.

Winkler, W.E., and L.R. Draper (1997), The SPEER Edit System. In: *Statistical Data Editing, Volume No. 2: Methods and Techniques* (pp. 51-55), United Nations, New York and Geneva.

Appendix: Proof of Theorem 2

41. To facilitate the proof of Theorem 2, it is convenient to first prove an auxiliary lemma. Define, for each edit ψ_q^k, the index set A_q^k of the edit(s) in the previous node from which it has been derived. That is to say, define $A_q^k := \{l\}$ if ψ_q^k is obtained by copying or fixing the value of a variable in edit ψ_{q-1}^l, and define $A_q^k := \{s,t\}$ if ψ_q^k is obtained by eliminating a variable from the pair of edits $\psi_{q-1}^s, \psi_{q-1}^t$.

Lemma 1. *Consider the situation sketched in Theorem 2, and suppose that x_g has been eliminated to obtain Ψ_q from Ψ_{q-1}. If A is a representing set of the index sets A_q^k that belong to all $\psi_q^k \in \Psi_q^{(2)}$, then there exists a value for x_g which, together with the original values of the variables that are involved in Ψ_q, satisfies all edits in Ψ_{q-1} except those in A.*

Proof. Because of the fundamental property of Fourier-Motzkin elimination, it is clearly possible to find a value for x_g that satisfies all pairs of edits from Ψ_{q-1} that lead to implied edits in $\Psi_q^{(1)}$. In addition, by construction A contains all indices of failed edits from Ψ_{q-1} that do not involve x_g, as required. The only way for the lemma to be false, would be if there existed two edits, say ψ_{q-1}^s and ψ_{q-1}^t, such that $s \notin A$ and $t \notin A$, for which it is not possible to find a value for x_g that satisfies both edits. In this case, an implied edit can be generated by eliminating x_g from ψ_{q-1}^s and ψ_{q-1}^t. Moreover, this implied edit must be failed by the original values of the other variables, by the fundamental property of Fourier-Motzkin elimination. In other words: the implied edit must be an element of $\Psi_q^{(2)}$. But this would contradict the assumption that A is a representing set of A_q^k for all $\psi_q^k \in \Psi_q^{(2)}$. This completes the proof of Lemma 1.

42. The proof of Theorem 2 now proceeds by induction on the number of treated variables q. For $q = 0$ the statement is trivial, and for $q = 1$ the theorem follows as a special case of Lemma 1. (Note that $B_1^k \equiv A_1^k$.) Suppose therefore that the statement has been proved for all $q \in \{0,1,\ldots,Q-1\}$, and consider the case $q = Q$, where $Q \geq 2$. If Ψ_Q is obtained from Ψ_{Q-1} by fixing a variable to its original value, and B is a representing set of the sets B_Q^k for the failed edits from Ψ_Q, then by construction B is also a representing set of the sets B_{Q-1}^k for the failed edits from Ψ_{Q-1}. Thus the statement for $q = Q$ follows immediately from the induction hypothesis in this case.

43. Suppose now that Ψ_Q is obtained by eliminating a variable, say x_g, from Ψ_{Q-1}. Define, for each $\psi_Q^k \in \Psi_Q^{(2)}$, the index set A_Q^k of (one or two) edits from Ψ_{Q-1} from which ψ_Q^k is derived, just as above. Next, use B to construct a set A through the following procedure, for each $\psi_Q^k \in \Psi_Q^{(2)}$:

- If ψ_Q^k was obtained by copying ψ_{Q-1}^l (so $A_Q^k = \{l\}$ and $B_Q^k = B_{Q-1}^l$), then add l to A.
- If ψ_Q^k was obtained by eliminating x_g from ψ_{Q-1}^s and ψ_{Q-1}^t (so $A_Q^k = \{s,t\}$ and $B_Q^k = B_{Q-1}^s \cup B_{Q-1}^t$), then add s to A if B contains an element of B_{Q-1}^s, and add t to A otherwise.

It is easy to see that this construction leads to a representing set A of the index sets A_Q^k for all $\psi_Q^k \in \Psi_Q^{(2)}$.

44. According to Lemma 1, there exists a value for x_g which, together with the original values of the variables that have not been treated, satisfies the edits in Ψ_{Q-1} except those in A. That is to say, Ψ_{Q-1} can be partitioned similarly to Ψ_Q as $\Psi_{Q-1} = \Psi_{Q-1}^{(1)} \cup \Psi_{Q-1}^{(2)}$, where $\Psi_{Q-1}^{(2)}$ contains the edits with indices in A. In addition, it is not difficult to see that the above construction implies that B is a representing set of the index sets B_{Q-1}^k for all $\psi_{Q-1}^k \in \Psi_{Q-1}^{(2)}$. Hence it follows from the induction hypothesis that, given the

original values of the variables that have not been eliminated *and* given the chosen value for x_g, there exist values for the other eliminated variables that satisfy all the original edits except those in B. This proves the statement for $q = Q$, and the proof of Theorem 2 is complete.