WP.21
ENGLISH ONLY

UNITED NATIONS STATISTICAL COMMISSION and ECONOMIC COMMISSION FOR EUROPE
CONFERENCE OF EUROPEAN STATISTICIANS

EUROPEAN COMMISSION
STATISTICAL OFFICE OF THE EUROPEAN COMMUNITIES (EUROSTAT)

Joint UNECE/Eurostat work session on statistical data confidentiality
(Manchester, United Kingdom, 17-19 December 2007)

Topic (ii): Tabular data protection

CELL SUPPRESSION IN A SPECIAL CLASS OF LINKED TABLES

Supporting Paper

Prepared by Peter-Paul de Wolf, Statistics Netherlands

Cell suppression in a special class of linked tables

Peter-Paul de Wolf*

* Statistics Netherlands, Department of Methodology and Quality, P.O. Box 4000, 2270 JM Voorburg, The Netherlands. (pwof@cbs.nl)

Abstract. A heuristic approach to protect a 'stand alone' hierarchical table has already been available for some time now and is known as either HiTaS or the modular approach. At NSI's often linked (hierarchical) tables need to be dealt with. To be able to protect linked tables, one first needs a way to easily define those tables. In the current paper we will discuss a possible way to describe a simple class of linked tables that is often considers at NSI's. Moreover, we will point out some irregularities that should be kept in mind, if such a method would be implemented.

1 Introduction

Cell suppression is an often used disclosure control technique to protect tabular data at NSI's. To use that technique on a single (hierarchical) table, several methods are available. E.g., in τ-ARGUS both the modular approach and the HyperCube approach can deal with hierarchical tables. The modular approach however, is not able to deal with linked tables. To extend the modular approach to generally linked tables is very difficult.

However, NSI's often have to deal with a very special kind of linked tables. E.g., they often publish a table with turnover specified by a detailed NACE code and aggregated Regional code as well as a table with turnover specified by a detailed Regional code and aggregated NACE code, while they won't publish a table with turnover specified by detailed NACE code and detailed Regional code.

A straightforward way to deal with those linked tables, is to protect the complete hierarchical table with both detailed NACE code as well as the detailed Regional code. However, this will often lead to over suppression: unsafe cells at the lowest level (detailed NACE and detailed Region at the same time) will often lead to secondary suppressions at the higher levels.

The heuristic HiTaS (also known as the modular approach) is such that it will protect all possible non-hierarchical subtables of a hierarchical table in a special order. See e.g., De Wolf (2002). Hence, it seems reasonable to extend this heuristic in the sense that it is allowed to discard certain subtables that will not be published

anyway. However, this means that we need a way to tell HiTaS which subtables need to be protected and which can be discarded with respect to disclosure control.

In the current paper we will suggest a simple way to define the special class of linked tables that was mentioned above. Section 2 contains some definitions that we will need. In section 3 we will present some examples. Since heuristics are used, the protection of the linked tables is not necessarily watertight. In section 4 we will show some problems that might arise.

2 Hierarchies

In this section we will define our notion of hierarchies, as we will be using throughout the paper. First we will need some general definitions, taken from graph-theory.

Definition 1 *A tree is an undirected graph T in which any two vertices is connected by exactly one path.*

Definition 2 *The distance between two vertices v and w is the number of edges on the path between v and w.*

Definition 3 *A directed tree with root r is a directed graph T which would be a tree if the directions on the edges were ignored and in which each vertex $v \neq r$ in T has a path from r to v.*

Definition 4 *In a directed tree with root r, a vertex v is a ancestor of vertex w if a path exists from v to w. Moreover, w is then called a descendant of v. In case a path of length one exists from v to w, then v is the father of w and w is a child of v.*

In the remainder of this paper, we will consider a hierarchy to be a rooted, directed tree, with the categories being the vertices of the tree. Using this representation, we can define the levels of the hierarchy as well

Definition 5 *The level in a hierarchy of a category, is the distance between the corresponding vertex to the root of the directed tree.*

Note that, by definition, the root category is at level 0.

In order to be able to compare different hierarchies when considering linked tables, we will need some more definitions.

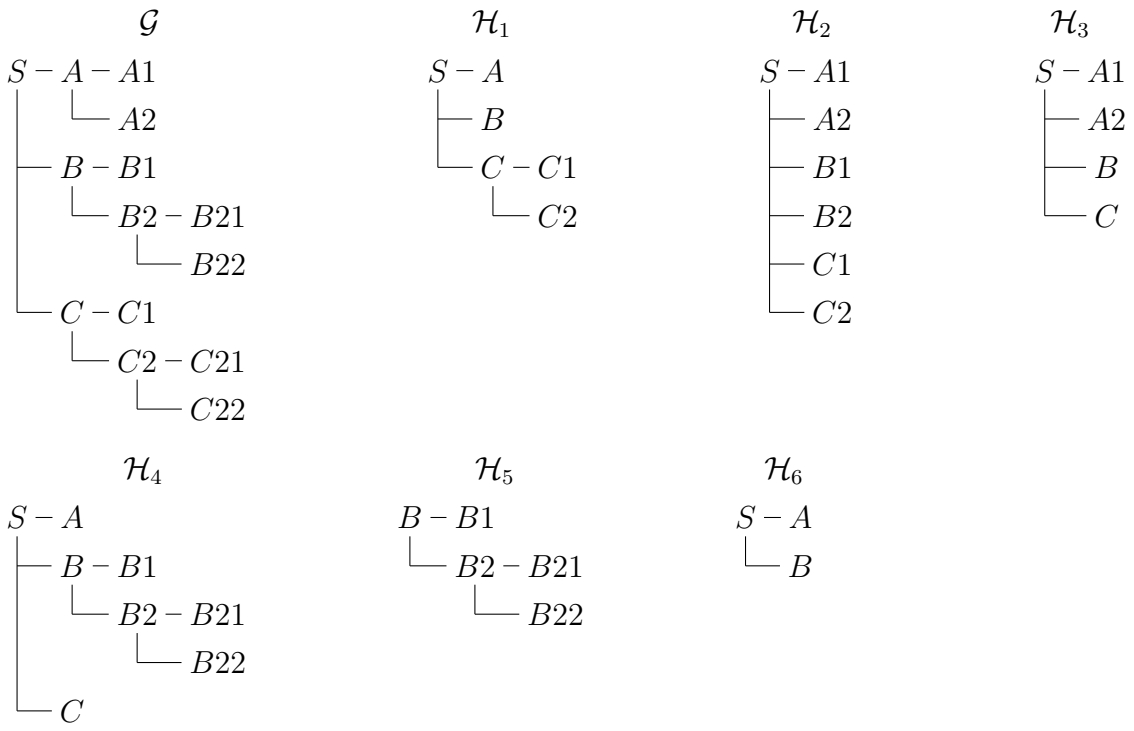
Definition 6 *A hierarchy $\mathcal{H}$ is called a pure sub hierarchy of hierarchy $\mathcal{G}$ if each path from vertex v to vertex w in $\mathcal{H}$ is also a path from v to w in $\mathcal{G}$. Notation: $\mathcal{H} \preceq \mathcal{G}$.*

Definition 7 *A set of hierarchies $(\mathcal{H}_1, \ldots, \mathcal{H}_K)$ is covered by hierarchy $\mathcal{G}$, if $\mathcal{G}$ is the hierarchy with the smallest number of vertices for which $\mathcal{H}_i \preceq \mathcal{G}$ for each $i = 1, \ldots, K$. Hierarchy $\mathcal{G}$ is then called the covering hierarchy.*

In the previous definition a covering hierarchy was defined for a set of hierarchies. In practice, we will often start with a detailed hierarchy and construct pure sub hierarchies by deleting all descendants of certain vertices. This leads to the following definition:

Definition 8 *A hierarchy $\mathcal{G}$ is a base hierarchy of hierarchy $\mathcal{H}$, in case $\mathcal{H}$ can be constructed by deleting all descendants of certain vertices in $\mathcal{G}$.*

A base hierarchy can be chosen such that it is a covering hierarchy as well. However, a covering hierarchy is not necessarily the (covering) base hierarchy of the same set of sub hierarchies. E.g., consider the following seven hierarchies:



We then have that

$$\mathcal{H}_1 \preceq \mathcal{G}, \qquad \mathcal{H}_2 \not\preceq \mathcal{G}, \qquad \mathcal{H}_3 \not\preceq \mathcal{G}, \qquad \mathcal{H}_4 \preceq \mathcal{G}, \qquad \mathcal{H}_5 \preceq \mathcal{G} \quad \text{and} \quad \mathcal{H}_6 \not\preceq \mathcal{G}.$$

Hierarchy $\mathcal{H}_6$ is not a pure sub hierarchy of $\mathcal{G}$, since root S of $\mathcal{H}_6$ equals root S of $\mathcal{G}$ if and only if the categories A and B differ from the categories A and B in $\mathcal{G}$.

Note that $\mathcal{G}$ is not a covering hierarchy of the set $(\mathcal{H}_1, \mathcal{H}_4, \mathcal{H}_5)$: vertices $A1, A2, C21$ and $C22$ of $\mathcal{G}$ are not needed to make $\mathcal{H}_1$, $\mathcal{H}_4$ and $\mathcal{H}_5$ pure sub hierarchies of $\mathcal{G}$. Additionally, $\mathcal{G}$ can be a base hierarchy of $\mathcal{H}_1$ and $\mathcal{H}_4$ but not of $\mathcal{H}_5$.

3 Example of linked tables

In this section we will provide an example of how to specify a set of linked tables and to derive a convenient cover table.

A user specifies N tables$T_1, \ldots, T_N$ that need to be protected simultaneously. Each table can have a hierarchical structure that may differ from the other hierarchical structures. However, tables that use the same spanning variables are only allowed to have hierarchies that can be covered.

Suppose that the specified tables contain M different spanning variables. Since the hierarchies are supposed to be coverable, an M-dimensional table exists having all the specified tables as subtables. The spanning variables will be numbered 1 up to M. The order only affects the way the M-dimensional table will be specified, not the way the suppression pattern is constructed.

Each spanning variable can have several hierarchies in the specified tables. Denote those hierarchies for spanning variable i by $\mathcal{H}_{i,1}, \ldots, \mathcal{H}_{i,\mathcal{I}_i}$ where $\mathcal{I}_i$ the number of different hierarchies.

Define the M-dimensional table by the table with spanning variables according to hierarchies $\mathcal{G}_1, \ldots, \mathcal{G}_M$ such that, for each $i = 1, \ldots, M$ hierarchy $\mathcal{G}_i$ covers the set of hierarchies $(\mathcal{H}_{i,j})$ with $j = 1, \ldots, \mathcal{I}_i$. This M-dimensional table will be called the cover table.

The modular approach (HiTaS) can now easily be adapted. HiTaS deals with all possible non-hierarchical subtables of a hierarchical table in a specially ordered way. We could keep this subdivision, but only consider those subtables that are also subtables of at least one of the specified tables $T_1, \ldots, T_N$ and disregard the other subtables.

Example 1 A user specifies three tables: T_1 with spanning variables $(S \times W \times R)$, T_2 with $(S \times G)$ and T_3 with $(S \times R)$. A base hierarchy for spanning variable S is given in figure 1.

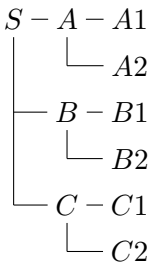


Figure 1: Base hierarchy $\mathcal{G}_S$ of spanning variable S

Since there are 4 different spanning variables, the cover table will be a four dimensional table. In figures 2–4 the hierarchies of the spanning variables are given, where the spanning variables S, R, G and W are numbered 1, 2, 3 and 4 respectively.

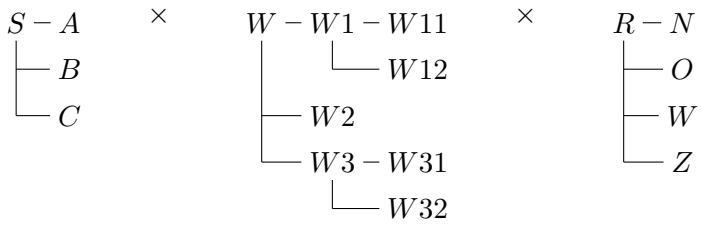


Figure 2: Table T_1: $\mathcal{H}_{1,1} \times \mathcal{H}_{4,1} \times \mathcal{H}_{2,1}$

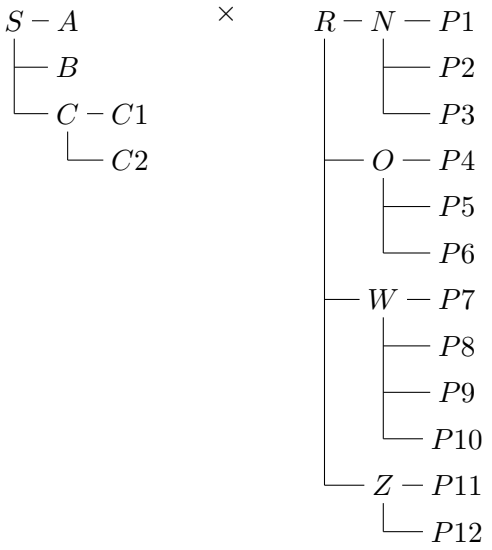


Figure 3: Table T_2: $\mathcal{H}_{1,2} \times \mathcal{H}_{2,2}$

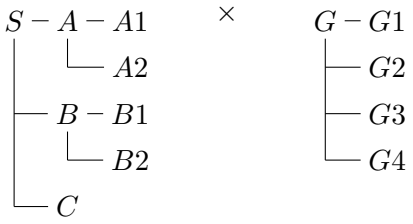


Figure 4: Table T_3: $\mathcal{H}_{1,3} \times \mathcal{H}_{3,1}$

The resulting 4-dimensional cover table will have the hierarchies $\mathcal{G}_1 = \mathcal{G}_S$ where $\mathcal{G}_S$ is the base hierarchy as given in figure 1, $\mathcal{G}_2 = \mathcal{H}_{2,2}$, $\mathcal{G}_3 = \mathcal{H}_{3,1}$ and $\mathcal{G}_4 = \mathcal{H}_{4,1}$. ◇

4 Problems

The idea of disregarding certain subtables may lead to situations in which the disclosure control is not necessarily watertight. In this section we will give two instances of problems that may arise.

In certain situations, the by the user specified tables may completely fix the internal structure of the higher dimensional table that will be considered. In case that internal structure is not included in the specified tables, it will not be considered when deriving a suppression pattern. Any, *implicitly* defined table structure will not be protected *explicitly*. The following example will show that in that way, implicitly a primary unsafe cell may be published.

Example 2 In figure 5 three tables are given that are specified by the user. These tables

	$B1$	$B2$	$B3$	Total
$A1$	30	150	70	250
$A2$	270	110	255	635
Total	300	260	325	885

	$C1$	$C2$	$C3$	Total
$A1$	140	50	60	250
$A2$	85	210	340	635
Total	225	260	400	885

	$C1$	$C2$	$C3$	Total
$B1$	110	90	100	300
$B2$	40	50	170	260
$B3$	75	120	130	325
Total	225	260	400	885

Figure 5: Three linked 2-dimensional tables

are considered to be safe on their own. However, they are the marginals of a three dimensional table $A \times B \times C$. Under the condition that all cells in that three dimensional table are non-negative, exactly one interior of that table is possible (see figure 6).

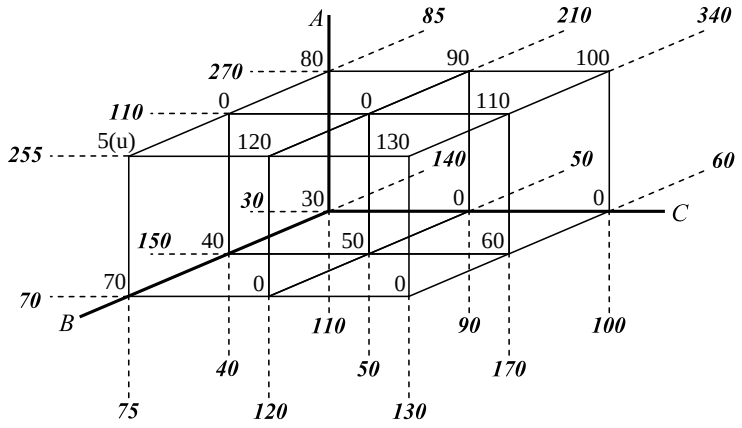


Figure 6: Graphical representation of $A \times B \times C$ of Example 2

So, if e.g., cell $(A2, B3, C1)$ would be primary unsafe (in the figure denoted by 'u'), the publication of the three 2-dimensional tables from figure 5 would hence implicitly lead to disclosure of a primary unsafe cell. ◇

Similarly, a suppression pattern can in certain situations be broken, due to the fact that the interior of the corresponding higher dimensional table is unique. See the following example.

Example 3 Figure 7 shows three tables with a possible suppression pattern. The bold cross is supposed to be a primary unsafe cell, the other crosses are secondary suppressions. Again, the underlying three dimensional table appears to be unique. See figure 8. Due to the uniqueness of the three dimensional table, the suppressed cells can easily be disclosed. ◇

	$B1$	$B2$	$B3$	Total
$A1$	10	50	x	x
$A2$	110	70	270	450
$A3$	110	250	x	x
Total	230	370	450	1050

	$C1$	$C2$	$C3$	Total
$A1$	50	20	**x**	x
$A2$	80	140	230	450
$A3$	250	120	x	x
Total	380	280	390	1050

	$C1$	$C2$	$C3$	Total
$B1$	120	50	60	230
$B2$	–	140	230	370
$B3$	260	90	100	450
Total	380	280	390	1050

Figure 7: Three linked 2-dimensional tables with suppression patterns

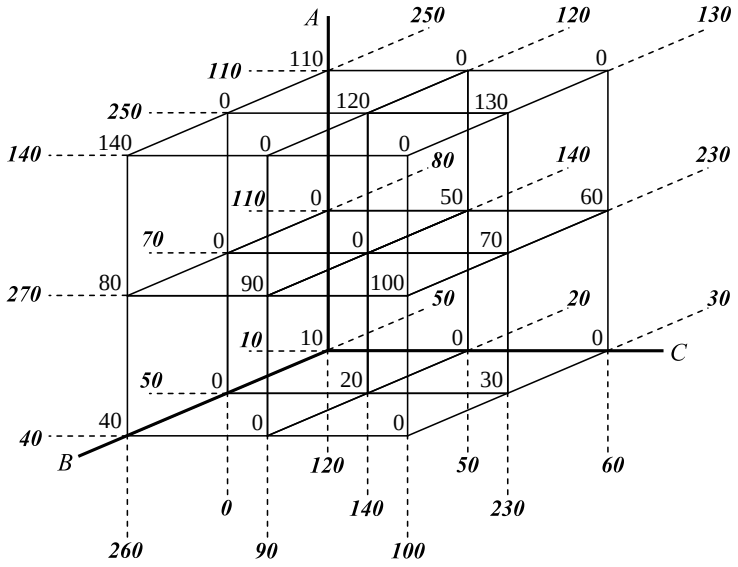


Figure 8: Graphical representation of table $A \times B \times C$ of Example 3

References

de Wolf, P.P. (2002). HiTaS: A heuristic approach to cell suppression in hierarchical tables. In J. Doming-Ferrer (Ed.), *Inference Control in Statistical Databases*, Berlin Heidelberg, pp. 74–82. Springer-Verlag.