Developments in Disclosure-Based Cell Suppression in Frequency Tables

Daniel P. Lupp and Øyvind Langsrud

Statistics Norway

Abstract. Frequency tables are a common way for national statistics institutes to publish information. One of the most common methods employed to protect against unwanted disclosure is cell suppression, where sensitive cell values are not published. A common heuristic used is the small count heuristic, where counts beneath a given threshold are considered sensitive. It is well-discussed in the community that this heuristic does not always protect tables well. In this work, we describe recent work done on defining a disclosure-focused approach to cell suppression.

Keywords: statistical disclosure control, confidentiality, cell suppression, official statistics, frequency tables

1 Introduction

Small count suppression is one of the predominant heuristics used when employing cell suppression for protecting frequency tables against disclosure. However, though it can be sufficient protection, this is not always the case. Indeed, cell suppression methods focus on protecting against disclosure of cell values. For magnitude tables, where the value or a unit's contribution to the value needs to be protected, this makes perfect sense. In frequency tables, however, an attacker is seldom interested in discovering a unit's contribution (as this is usually 1). The type of disclosure one usually tries to protect against in frequency tables is categorical attribute disclosure, which occurs when a unit's membership in certain cells can be deduced [1]. For this type of disclosure, simply suppressing small counts is insufficient. Consider the following rows of a table containing traffic-related accidents in Norwegian cities:

	Uninjured	Light	Serious	**Total**
Oslo	0	0	30	**30**
Bergen	0	0	2	**2**

Though each row has vastly different values, both are equally disclosive in the above sense: an attacker can deduce, without further knowledge about contributing units, that all units in Oslo (resp. Bergen) were seriously injured. Applying the small count suppression heuristic (without suppression of zeros) results in the following table:

	Uninjured	Light	Serious	**Total**
Oslo	0	0	30	**30**
Bergen	0	0	×	×

Here, no suppressed cell value can be recalculated (indeed, the estimate interval is infinite), yet the suppression can hardly be labeled a success: The table is as disclosive as it was pre-suppression. Assuming one knows of a unit in Bergen (resp. Oslo), then it must have been seriously injured. Indeed, the disclosure occurs independently of single table cells' value, as witnessed in the example table above. For this reason, some publishing institutions have adopted the practice of suppressing all zeros as well. In [3], the authors argue this can, depending on the table, lead to over or under-protection (in other words, still does not address the disclosure), and propose a new heuristic aimed at targeting such disclosure directly. In this work, we give a brief description of that method and describe some of its weaknesses in Section 2. In Section 3, we provide an overview of recent advances and a more general methodology. An article containing a complete description of the new method, including details related to implementation, is in preparation for submission to a journal. A draft version of this paper is available on the arXiv [4].

2 Direct Disclosure

This section discusses the method described in [3]. The paper proposes a formalization of an attacker's knowledge, thus defining when disclosure happens, as well as propose a new scheme for cell suppression. The work focuses on individual and group attribute disclosure, in particular what is referred to as "within-group disclosure" in [1, p. 187]: when a unit in a group can infer more information about other units in the same group. For example, the first row in Table 1 (left) illustrates within-group disclosure, where the lightly injured unit can infer that all other units in Oslo are seriously injured. [3] generalizes this notion by formalizing an attacker's knowledge: an attacker A is a *k-coalition*, i.e., a group of k units contributing to the table. Thus, an attacker can use its full knowledge of k units to disclose information about other units. A cell c is then called *directly disclosive w.r.t. k* if it contributes to a published marginal p, and the difference $\text{freq}(p) - \text{freq}(c)$ is less than or equal to k. In practice, the cell with the largest contribution to the marginal will be directly disclosive. The authors make a further distinction: not all cell membership disclosures represent an attacker gaining knowledge. Indeed, unknown categories can play a specific role in protecting the table, thus reducing the need for cell suppression.

Unknown categories [3] believe that not all such direct disclosures necessarily need protection, and that this should be up to the publishing entity's policies. In particular, one can differentiate between different types of unknown categories: *disclosive* and *non-disclosive unknowns*. These are related to the following questions [3]:

	Uninj.	Light	Serious	**Total**
Oslo	0	1	5	**6**
Trondheim	4	1	3	**8**
Bergen	0	0	2	**2**
Stavanger	1	4	4	**9**

	Uninj.	Light	Serious	**Total**
Oslo	0	+	×	**6**
Trondheim	4	1	3	**8**
Bergen	0	+	×	**2**
Stavanger	1	4	4	**9**

Table 1. Example of a frequency table (left) summarizing levels of injury in traffic accidents in Norwegian cities, with the direct-disclosure suppression for $k = 1$ on the right. × and + represent primary and secondary suppressions respectively.

1. Does the disclosure of an unknown value lead to an attacker learning information about a statistical unit?
2. Can an unknown value be used to identify oneself or others in the table?

For example, for a location variable `Region`, it is reasonable to assume that the value "Unknown" would lead to the answer "no" for both of the above questions. Therefore, disclosures of "Unknown" and disclosures requiring an attacker to know which units have "Unknown" in the `Region` variable need no protection. Not all unknown values, however, have this property. Consider "Unknown disease" in a frequency table describing medical information. This can be used to infer that a unit is sick, thus answering "yes" to the first question. Furthermore, depending on the specific situation, this value could be used to identify oneself or others in a table: it can happen that someone knows they have an unknown disease. For these types of unknowns, disclosures need to be protected.

Direct disclosure suppression In the paper [3], the authors suggest a new heuristic for cell suppression in order to protect against direct disclosure. Here, the directly-disclosive cells not protected by unknown categories are identified and primary suppressed. Then, in order to ensure no direct disclosure in the fully suppressed table, secondary suppressed cells must be chosen carefully, and zeros must be permitted as candidates for suppression. An example of such suppression can be seen in Table 1.

Example 1. Consider the first row in Table 1 (left). Here, a 1-coalition consisting of the lightly injured unit in Oslo can disclose that all other units in Oslo are seriously injured. Now consider the following alternative suppression patterns which attempt to protect against this disclosure:

	Uninj.	Light	Serious	**Total**
Oslo1	0	1	×	×
Oslo2	×	×	5	**6**
Oslo3	0	×	×	**6**
Oslo4	×	1	×	**6**

In row Oslo1, the 1-coalition can see that all other units must be seriously injured. Note that the attacker need not know the value of the (Oslo, Serious) cell in order to make this disclosure. In row Oslo2, the 1-coalition comes to

the same conclusion: it knows that all units contributing to suppressed cells are members of the coalition (i.e., in (Oslo, Light)), and that therefore there are 1) no uninjured units and 2) only one lightly injured unit. It is worth noting that this suppression pattern (and thus this disclosure) would occur when applying the standard of suppressing small counts and zeros.

In row Oslo3, the attacker cannot disclose the level of injury of the other units, since it does not know how the suppressed 5 units are distributed between Light and Serious. Similarly, in Oslo4 the 1-coalition can see that there are no other lightly injured units, but cannot say how the remaining 5 units are distributed between uninjured and seriously injured.

The suppression method in [3] shows promise for simple tables. However, there are limitations. On the one hand, choosing the right secondary suppressions to prevent disclosure is a complex task. Secondary cells must be chosen not only to prevent recalculation of the sensitive cells, but additionally guarantee that no direct disclosure occurs. Thus, the problem formulation is different than the original cell suppression problem (CSP), and finding an optimal solution would require at the very least considerable adjustment of existing solutions. Furthermore, disclosure of combinations of cells not represented as published marginals cannot be protected. Consider the righthand table in Table 1. In this table, though no single unit can deduce cell membership of another unit (i.e., it is protected from direct disclosure), the table still contains a disclosure worth protecting: in Stavanger, the uninjured unit can disclose that all other units are injured, i.e., lightly or seriously injured. The premise that directly-disclosive cells must be primary suppressed makes it hard to generalize the method to more complex scenarios: if one wishes to protect against membership disclosure for such groups of cells (e.g., Light and Serious as Injured), which of these cells should be primary suppressed? In the next section, we describe a more general approach to disclosure-driven suppression, called k-disclosure [4]. This method provides a more robust handling of disclosure protection. In particular, the problem of suppressing to protect against k-disclosure reduces to classical CSP, thus allowing the use of existing solvers with some adjustments. Furthermore, it supports protection of combinations of cells in a natural way.

3 k-Disclosure

This section describes a generalization of direct disclosure. A full description of the method can be found in [4]. The premise remains the same as in direct disclosure: we assume an attacker to have full knowledge of up to k units, i.e., an attacker is a k-coalition. However, we wish to extend the protection to include disclosure of groups of cells, called *meaningful combinations*. The method is based on the following principle:

Principle 1 *Let A be a collection of units contributing to a frequency table T. Then A can disclose membership of a unit i in a group of cells C if and only if*

1. *A knows that A and i both contribute to some sub-population P of units in T,*
2. *A can deduce that all units in P are in A or C.*

As compared with direct disclosure, this is a generalization as it allows for groups of cells C. Furthermore, since P is simply described as a "sub-population", it need not refer to a published marginal. This allows for further flexibility, as we discus in the following Section 3.1. Similarly to direct disclosure, these two points are fulfilled if and only if the difference between P and C is less than or equal to the size of A (k, if A is a k-coalition). The key insight of the new method is the choice of primary cells: Since the differences are what lead to disclosure, they must be considered sensitive. In the terms of cell suppression, one wishes to prevent recalculation of these sensitive differences, i.e., they must be primary suppressed. Then secondary cells must be chosen in such a manner that the estimate interval of the differences includes numbers greater than k (thus ensuring that Principle 1 does not hold). However, as such differences are seldom represented as a cell to be published, this requires an adaptation to the classical cell suppression problem.

A naive approach to supporting such functionality would be to include all differences between marginals and their contributing cells in the table and primary suppressing those difference cells that lead to disclosure.[1] This would result in a considerable increase in table size, greatly increasing run time of the suppression algorithm. Instead, we suggest pre-processing the table and adding only those difference cells that lead to disclosure (and, for example, possibly taking non-disclosive unknowns into consideration). Then the secondary suppression algorithm may choose as candidates only the table cells meant to be published.

3.1 Meaningful Combinations

The k-disclosure method adds support for specifying groups of cells that need protection, called meaningful combinations. This also provides functionality for protecting against user-defined negative disclosure, where an attacker can deduce that a unit is *not* in a certain group of cells. In this context, disclosure can naturally be divided into two categories: disclosure *of* meaningful combinations, and disclosure *within* meaningful combinations.

Disclosure of meaningful combinations This form of disclosure refers to an attacker disclosing another units membership in a meaningful combinations. Consider Table 2 (left): In the first row, an attacker without any knowledge about statistical units (a 0-coalition) can deduce that all units in Oslo were injured. Likewise, in the fourth row, an attacker with knowledge of the uninjured unit in Stavanger (a 1-coalition) can disclose that all other units are injured. In order

[1] Note that, when we refer to adding difference cells, we do not mean find the group of cells that represent the difference. Rather, we mean create a new cell that describes the structure represented by the difference.

	Uninj.	Light	Serious	**Total**
Oslo	0	1	5	**6**
Trondheim	4	1	3	**8**
Bergen	0	0	2	**2**
Stavanger	1	4	4	**9**

	Uninj.	Light	Serious	**Total**
Oslo	+	1	+	**6**
Trondheim	4	1	3	**8**
Bergen	+	0	+	**2**
Stavanger	+	4	+	**9**

Table 2. The same table as in Table 1 (left), a 1-disclosure suppression with meaningful combination Injured = Serious + Light, without control of "disclosure within" (right). Since the primary difference cells are hidden, all suppressed cells (+) are secondary suppressions.

to protect against these disclosures, one can apply Principle 1: the attackers can deduce that units are injured in Oslo (resp. Stavanger), because they see that the difference between the common sub-population (in this case, the row totals) and the units in the meaningful combination (the sum of Light and Serious) is less than the coalition size. In order to protect against such disclosure, we can add these difference cells to our cell suppression problem, primary suppress them, and continue as described in the previous section. The result of this can be seen in Table 2 (right).

Disclosure within meaningful combinations This form of disclosure happens when an attacker does not need a published marginal to represent the common sub-population to disclose a unit's attribute. Consider the second row in Table 2 (left). Here, an attacker with knowledge of the lightly injured unit in Trondheim (a 1-coalition) can deduce that all other injured units must be seriously injured. The publishing entity must decide whether or not it is likely that an attacker possesses such detailed information about other units. If they deem it too high a risk, then we once again can apply Principle 1 and the subsequent discussion. This time, however, the difference cell does not refer to the difference between a marginal and a sum of cells. Rather, one must add a cell representing $((Trondheim, Light) + (Trondheim, Serious)) - (Trondheim, Light)$, and then proceed with the cell suppression as described previously.

4 Implementation

Both methods, direct disclosure and k-disclosure suppression, have been implemented in the `GaussSuppression` R-package [2]. In particular, the internal structure used to describe suppression problems is well-suited for this approach.

The package uses a *model matrix* X to describe the relationship between the input y and the output z, such that $z = X^T y$, where y and z are vectors of frequencies. Here, each row of X corresponds to one entry of the input data (either an aggregated table or microdata), and each column represents a published cell. Thus, an entry x_{ij} in X is equal to 1 iff row i in the input contributes to published cell j. Then the secondary suppression algorithm uses Gaussian elim-

ination on X to determine which columns of X can lead to recalculation of the primary cells (also represented by columns in X).

Due to this structure, adding support for k-disclosure suppression was fairly straightforward: rather than having primary cells correspond to columns in X, one needed only extend the tool to allow for custom columns not contained in X to be considered primary suppressed. In that manner, one can define the new columns representing the (not to be published) difference cells by adding/subtracting the relevant columns of X and running the same algorithm.

A weakness of the Gaussian elimination method is the non-optimality with regards to number of suppressed cells and no control over estimate intervals. Protection against exact recalculation of suppressed frequencies is achieved, but currently the method has no control over what intervals an attacker can estimate for suppressed cells. As such, it is not guaranteed that the suppression will protect against disclosure. If, for example, a table is suppressed in such a way that a 3-coalition can estimate a difference to be in the interval $[0, 2]$, the disclosure can still occur according to Principle 1. Still, publishing entity's must weigh the risk with utility: the algorithm shows promising protection in practice, and allows for comparably quick suppression of very large tables.

We believe adding k-disclosure functionality is a candidate for future development in other popular tools, such as `tauArgus` [6] and `sdcTable` [5]. However, we are unsure as to whether or not this will be straightforward to implement. Even if the initial table can be defined by a single hierarchy-setup (tree-shaped), the table including differences may need to be specified as multiple linked tables. It is also advantageous to include only the differences that are primary suppressed (others are hidden and are not needed). Furthermore, zeros must be able to be secondary suppressed, though this problem might not prove difficult to solve for these tools. The benefit of implementing this functionality in the aforementioned tools would be to have access to more optimal secondary suppression algorithms, where one in addition can have some control over the attacker's intervals.

Acknowledgments We would like to thank Peter Paul de Wolf from Statistics Netherlands for his insightful questions at and after the UNECE Expert Meeting on SDC 2021, in particular those regarding meaningful combinations, which helped make this work more mature.

References

1. Hundepool, A., Domingo-Ferrer, J., Franconi, L., Giessing, S., Nordholt, E., Spicer, K., de Wolf, P.: Statistical Disclosure Control. Wiley Series in Survey Methodology, Wiley (2012)
2. Langsrud, Ø., Lupp, D.P.: GaussSuppression: Tabular Data Suppression using Gaussian Elimination (2022), https://CRAN.R-project.org/package=GaussSuppression, r package version 0.4.0
3. Lupp, D.P., Langsrud, Ø.: Suppression of Directly-Disclosive Cells in Frequency Tables (2021), Joint UNECE/Eurostat Expert Meeting on Statistical Data Confidentiality, 1–3 December 2021, Poznań, Poland

4. Lupp, D.P., Langsrud, Ø.: Disclosure-Driven Suppression in Frequency Tables (2022), URL to be updated, preprint.
5. Meindl, B.: sdcTable: Methods for Statistical Disclosure Control in Tabular Data (2021), https://CRAN.R-project.org/package=sdcTable, r package version 0.32.2
6. de Wolf, P.P., Hundepool, A., Giessing, S., Salazar, J.J., Castro, J.: tau-ARGUS user's manual, version 4.1. Tech. rep., Statistics Netherlands (2014)