UNECE

CONFERENCE OF EUROPEAN STATISTICIANS

Expert Meeting on Statistical Data Confidentiality

1-3 December 2021, Poland

Suppression of directly-disclosive cells in frequency tables.

Daniel Lupp (Statistics Norway)
daniel.lupp@ssb.no

Abstract

Recent law changes in Norway have forced Statistics Norway to re-evaluate the use of SDC methods for protecting its publications. Previous distinctions regarding the sensitivity of variables are no longer included in the law. Instead, publications are required by law to ensure that no identifiable individual can have attributes directly or indirectly disclosed.

Common heuristics (such as small count primary suppression) used in frequency table suppression do not directly target disclosure, but rather rely on established rule-of-thumb. Such methods can lead to the suppression of a large number of cells and simultaneously not provide adequate protection from disclosure. Motivated by risk measures within microdata, we discuss a new approach aimed directly at detecting those cells that can lead to disclosure in frequency tables. In this context we define the notion of direct disclosure: the scenario where an attacker, with full knowledge of themselves (or a coalition), can disclose attributes of specific individuals.

Finally, we present an implementation where the above method for primary suppression is combined with an alternative method to secondary suppression which aims to guarantee no linear dependency between published and suppressed cells. The secondary suppression method is implemented using Gaussian elimination and shows promising results in practice, striking a balance between computational complexity and number of suppressed cells. The presented methods are available in the R package GaussSuppression.

Suppression of Directly Disclosive Cells in Frequency Tables

Daniel P. Lupp*, Øyvind Langsrud**

* Statistics Norway, daniel.lupp@ssb.no

** Statistics Norway, oyvind.langsrud@ssb.no

Abstract. Recent law changes in Norway have forced Statistics Norway to re-evaluate the use of SDC methods for protecting its publications. Previous distinctions regarding the sensitivity of variables are no longer included in the law. Instead, publications are required by law to ensure that no identifiable individual can have individual attributes directly or indirectly disclosed. Common heuristics (such as small count primary suppression) used in frequency table suppression do not directly target disclosure, but rather rely on established rule-of-thumb. Such methods can lead to the suppression of a large number of cells and simultaneously not provide adequate protection from disclosure. Motivated by risk measures within microdata, we discuss a new approach aimed directly at detecting those cells that can lead to disclosure in frequency tables. In this context we define the notion of direct disclosure: the scenario where an attacker, with full knowledge of themselves (or a coalition), can disclose attributes of specific individuals. Finally, we present an implementation where the above method for primary suppression is combined with an alternative method to secondary suppression which aims to guarantee no linear dependency between published and suppressed cells. The secondary suppression method is implemented using Gaussian elimination and shows promising results in practice, striking a balance between computational complexity and number of suppressed cells. The presented methods are available in the R package GaussSuppression.

1 Introduction

Since January 2021, the Norwegian Statistics Act has been changed to more closely align with the laws on official statistics in the EU. To that effect, the wording in the section on confidentiality regarding the publication of official statistics has undergone various changes that have forced Statistics Norway to re-evaluate its use of SDC methods. The new law is as follows [4]:

> **§7 Statistical confidentiality in dissemination of official statistics**
> (1) Official statistics shall be disseminated in such a manner that it is not possible to directly or indirectly identify a statistical unit and thus disclose individual data.

Critically, the new phrasing removes a previous clause providing an exemption for unsensitive data. Statistics Norway does not, as of yet, have an official institutional

			Injury				
Region	Vehicle	TrafGr	None	Light	Serious	Unknown	Total
Oslo	Car	Driver	0	0	17	0	17
Oslo	Car	Passenger	0	3	8	0	11
Bergen	Car	Driver	1	0	12	0	13
Bergen	Car	Passenger	0	1	0	14	15
Trondheim	Bicycle	Driver	3	2	2	0	7

Table 1: Example frequency table.

interpretation of the law. However, there is an ongoing effort to check whether its publications are compliant with the new laws.

Common disclosure risk measures for magnitude tables, such as dominance and $p\%$ rules, aim at preventing disclosure by suppressing those cells where a contributor can too closely estimate other units' contributions [1]. Interestingly, this does not seem to be the case for common risk measures for frequency tables, such as small count suppression. This does not directly target the prevention of disclosure but is rather based on an established rule-of-thumb. This paper presents an alternative heuristic for primary suppression in frequency tables with dislosure explicitly in mind. In its most basic form, the proposed heuristic checks the frequency table for group disclosures, such as the first row in Table 1, with an approach similar to l-diversity [3]. Additionally, the method provides a means of encoding an attacker's background knowledge: given that an attacker has knowledge of themselves or a set of other table contributors, the method targets those cells they can use to disclose information about other statistical units. The intention is to provide a formalized understanding of (parts of) the new Norwegian Statistics Act and thus serve as a baseline protection for frequency tables.

2 Direct disclosure

In the following we introduce the property of *direct disclosure* for table cells. We begin by defining the scope of what we define as undesired disclosure. In Section 2.2, we provide a formal definition of direct disclosure given this legal interpretation. In Section 2.3, we argue that "unknown" values in variables deserve special treatment and can, by themselves, provide a form of protection from disclosure. In Section 2.4 we discuss how these pieces can be combined in order to suppress cells with the specific intention of preventing undesired disclosure.

As an abuse of nomenclature, we shall for the remainder of this paper occasionally refer to the set of cells consisting of a marginal and its contributing inner cells as a *row*. This is motivated by common visualizations of frequency tables and is solely intended to simplify discussion.

2.1 Legal interpretation

Our approach is based on some key assumptions which provide a formalized interpretation of the law. More concretely, one needs to define what "identify a statistical unit" and "disclose individual data" mean.

For the purposes of the presented heuristic, we understand identification of a statistical unit as an attacker being able to deduce sensitive information about specific statistical units. In practice, this reduces to preventing group disclosures given the attacker's knowledge. The phrase "disclose individual data" refers to the disclosure of sensitive information. As currently implemented, this is interpreted as the disclosure of *atomic values* (that is to say, singular values) in sensitive variables, i.e., negative or disjunctive disclosure is not considered sensitive. For example in Table 1, an attacker deducing a singular value in the `Injury` variable for a group ("every driver involved in a car crash in Oslo was seriously injured") is considered disclosure of sensitive information. On the other hand, disclosure that a group must have some value in a combination of values in the `Injury` variable ("every passenger involved in a car crash in Oslo was either seriously or lightly injured") is not considered disclosure of sensitive information.[1]

2.2 Directly disclosive cells

We assume that an attacker has knowledge of k contributors to the table (called a k-coalition). In the most basic case, an attacker has no background knowledge whatsoever, i.e., $k = 0$. However, one can often assume that most statistical units know themselves, and hence can place themselves in the correct cell, i.e., $k = 1$ (this is the default value for the R implementation of this method). An attacker can then use their background knowledge to disclose information about other units by removing themselves from the data and analyzing the resulting table. The aim of the presented method is to prevent such disclosure.

To that effect, we call a cell c in a frequency table *directly disclosive w.r.t. k*, if there exists a published marginal cell p_c within a sensitive variable, such that

1. c is a cell contributing to p_c, and
2. $\text{freq}(p_c) - k \leq \text{freq}(c)$.

In other words, if a cell is directly disclosive w.r.t. k then there exists an attacker with knowledge of k table contributors that can deduce that all other units contributing to p_c must be in the cell c. For example, consider once more Table 1. For $k = 0$, it contains one directly disclosive cell (first row): (Oslo, Car, Driver, Serious) with frequency 17. In other words, an attacker without any background information can use this cell to disclose information about a group. Assuming an attacker has knowledge of up to three table contributors, i.e., $k = 3$, there exist four directly disclosive cells within the sensitive variable `Injury`: (Oslo, Car, Driver, Serious) with frequency 17, (Oslo, Car, Passenger, Serious) with frequency 8, (Bergen,

[1] There is ongoing work on generalizing this in order to allow users to define their own interpretation of what (combinations of) values constitute undesired disclosure. However, this is not implemented in the current release.

Car, Driver, Serious) with frequency 12, and (Bergen, Car, Passenger, Unknown) with frequency 14. Thus contributors to these three cells are at risk of having their level of injury directly disclosed by such an attacker. This information can guide our primary suppression: by suppressing directly disclosive cells (and with adequate secondary suppression), one can prevent such dislosures. However, as we shall see in the following section, not all directly disclosive cells need to be suppressed: exceptions can be made under certain circumstances for the disclosure of cells corresponding to unknown values.

2.3 Disclosive and nondisclosive unknowns

In this section we define the notions of disclosive and nondisclosive unknowns. These play a major role when answering the following questions and are, as such, an important part of direct disclosure suppression:

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?

With regards to the first question, consider once again Table 1. According to the definition from the previous section, the cell containing car passengers in Bergen with unknown injury is directly disclosive: the lightly injured car passenger in Bergen can deduce that all other car passengers in Bergen have an unknown injury. Thus, the attacker has arguably not gained any further knowledge about other statistical units; rather, the attacker knows simply that the underlying data is incomplete. Thus, ”unknown injury” can be considered a *nondisclosive unknown value*: even if an attacker can deduce that a unit must contribute that cell, the attacker has not learned anything about that unit.

However, not all unknown values are nondisclosive; in certain cases a lack of knowledge is knowledge itself. Consider a scenario where a frequency table contains a variable for the self-reported citizenship of a parent/guardian. If an attacker can deduce that a unit has an unknown value in this variable, then the attacker has learned that this unit does not know its parent or guardian's citizenship. This is an example of *disclosive unknown values*.

Regarding the second question, consider the following row in a table:

			`Injury`				
`Region`	`Vehicle`	`TrafGr`	None	Light	Serious	Unknown	Total
Unknown	Car	Driver	1	0	7	0	8

Though this row exhibits direct disclosure of seriously injured car drivers, the `Region` category is a nondisclosive unknown and thus no attacker will be able to disclose information about specific units, since an attacker will not be able to place either

			Injury				
Region	Vehicle	TrafGr	None	Light	Serious	Unknown	Total
Oslo	Car	Driver	0	0	17	0	17
Bergen	Car	Driver	1	0	12	0	13

			Injury				
Region	Vehicle	TrafGr	None	Light	Serious	Unknown	Total
Oslo	Car	Driver	0	0	X	0	X
Bergen	Car	Driver	X	X	12	0	13

			Injury				
Region	Vehicle	TrafGr	None	Light	Serious	Unknown	Total
Oslo	Car	Driver	0	X	X	0	17
Bergen	Car	Driver	X	0	X	0	13

Figure 1: The choice of secondary suppression matters to prevent direct disclosure. The first table contains original values, the second table contains valid secondary suppression where cell values cannot be recalculated but where information is still disclosed, and the last table contains suppressed cells that prevent disclosure.

themselves or other units in these cells. In other words, if a variable has a nondisclosive unknown value, it cannot be used as a method of identification, and hence the corresponding rows do not need primary protection.

In the R implementation of direct disclosure suppression in [5], all categories in sensitive variables are considered disclosive by default. When setting up the suppression method, the domain expert or methodologist can decide whether any sensitive variable contains a nondisclosive unknown value and pass this as a parameter. This will greatly depend on the domain in question and should be considered carefully.

2.4 Suppressing directly disclosive cells

Choosing suitable cells during suppression is crucial for preventing direct disclosure. Consider Fig. 1, where the first table depicts two rows from our running example that exhibit directly disclosive cells for $k = 1$. The second table in Fig. 1 illustrates two important points:

1. Primary suppressing the cells an attacker has knowledge about does not protect from direct disclosure unless the disclosed cell is also suppressed. For $k = 1$, an attacker in the second table of Fig. 1 with knowledge about the uninjured car driver in Bergen can immediately see from the difference between the row-total and the maximum cell that all other car drivers in Bergen were seriously injured. Note that small-count primary suppression would result in this suppression pattern if zeros are allowed to be secondary suppressed.

2. Even when primary suppressing directly disclosive cells, a valid secondary suppression where cell values cannot be recalculated is not sufficient protection against direct disclosure; the choice of secondary cells is important. Consider the first row of the second table in Fig. 1 describing injury status of car drivers in Oslo. Here the actual value of the cell is irrelevant for disclosure: despite

Region	Vehicle	TrafGr	Injury				
			None	Light	Serious	Unknown	Total
Oslo	Car	Driver	s	0	p	0	17
Oslo	Car	Passenger	0	s	p	0	11
Bergen	Car	Driver	s	0	p	0	13
Bergen	Car	Passenger	0	1	0	14	15
Trondheim	Bicycle	Driver	3	2	2	0	7

Table 2: Suppressing directly disclosive cells for $k = 3$ and nondisclosive unknown in the `Injury` variable. Cells labeled p and s are primary and secondary suppressed, respectively.

an attacker not being able to recalculate the cell value, they can immediately deduce that all car drivers in Oslo involved in a crash were seriously injured.

The third table in Fig. 1 shows a valid secondary suppression that prevents disclosure of directly-disclosive cells for $k = 1$.

The R implementation of direct disclosure suppression is implemented in the function `SuppressDirectDisclosure` in the R package GaussSuppression [5]. This primary suppresses directly disclosive cells (omitting those which are "safe" with respect to nondisclosive unknowns), and runs a secondary suppression method which suppresses cells suitable for protecting against disclosure (see point 2 above). Table 2 shows the full direct disclosure suppression for the running example from Table 1.

The secondary suppression method is based on Gaussian elimination, which we introduce in the next section.

3 Secondary suppression via Gaussian elimination

The vector of all primary suppressed cells can be written as $z_P = X_P^T y$ where y is the vector of all inner cells (all variables crossed) and where X_P is a dummy matrix (0's and 1's). Similarly, the vector of all cells considered to be published can be written as $z_C = X_C^T y$. Within our framework, some of the cells in z_C need to be secondary suppressed whenever any of the columns of X_P linearly depends on X_C. If such a dependency exists, one can compute the value of an element of z_P as a linear combination of z_C.

The cells to be selected for secondary suppression are chosen according to a sequential algorithm. First, the columns of X_C are ordered according to some measure of importance with the most important first (normally the overall total). A matrix X_C^* is then composed by including columns from X_C, one by one, in the selected order. Each time a new column is included, the algorithm checks whether X_P linearly depends on the new candidate version X_C^*. If this is the case, the last column included is omitted and instead the corresponding value of z_C is set as secondary suppressed. When all columns of X_C are checked this way, the cells to be secondary suppressed are found.

In practice, an efficient algorithm that avoids repeating similar calculations should be used. Gram-Schmidt orthogonalization is one possibility. For large problems, sparse matrix methodology is needed and the algorithm should maintain sparsity as much as possible. For this reason, an algorithm based on Gaussian elimination has been implemented in the R-package SSBtools.

Efforts to handle the singleton problem and the related problem of zeros has been included in the algorithm. The algorithm does not solve the suppression problem in an optimal way, but the ability to select the order in which cells are considered for secondary suppression is a useful feature. This feature is used to ensure adequate secondary protection in direct disclosure suppression.

A problem to be aware of is when zero frequencies are primary suppressed or are candidates for secondary suppression. For the calculation of z_P and z_C it does not matter whether zeros are included in y or whether the rows are omitted completely from X_P, X_C and y. This does, however, matter for the algorithm. When zeros are (candidates to be) suppressed, the corresponding rows need to be included. This property can also be seen as a strength, since this means we have a way of dealing structural zeros. Structural zeros should not be included in y, but they can occur in z_C.

4 Concluding remarks

Frequency tables can be viewed as a compact representation of microdata with categorical variables. As such, we can see similarities between tabular risk measures and microdata risk measures. Indeed, direct disclosure can be viewed as a nuanced variant of l-diversity. In the simplest case, i.e., for $k = 0$ and no nondisclosive unknowns, directly disclosive cells occur precisely in rows that breach the 2-diversity condition. For $k > 0$, it becomes 2-diversity with added conditions regarding the counts of distinct values in sensitive variables. This can not only be used to distinguish methods, the direct disclosure approach can thus also provide risk measures for microdata. This is similar to the risk/utility considerations made in [2].

It is important to note that we do not claim that direct disclosure suppression is the end-all method for protecting against disclosure in frequency tables. It is intended to provide an interface which allows for a formalized understanding of the law, by providing the ability to specify which variables and values should be protected from disclosure. It is intended as a baseline for detecting the most rudimentary disclosures, and can (and should possibly) be combined with other means of protection based on the publishing institution's understanding of the law. Indeed, the package GaussSuppression has functionality for combining multiple (also custom) primary suppression functions with this in mind.

The default implementation in R represents a rather strict interpretation of the law: each variable is both quasi-identifying and sensitive, and all values are disclosive, though this can be tailored to one's needs. The code is still under active development, and as such does not contain all the desired features. For instance, the version on CRAN does not yet support hierarchical variables. Another feature to be added in the foreseeable future are the ability to customize what disclosures are considered sensitive, e.g., combinantions of values might be sensitive or specific values might be considered non-sensitive.

References

[1] Anco Hundepool, Josep Domingo-Ferrer, Luisa Franconi, Sarah Giessing, Eric Schulte Nordholt, Keith Spicer, and Peter-Paul de Wolf. *Statistical Disclosure Control.* Wiley, 2012.

[2] Øyvind Langsrud and Daniel P. Lupp. Fair risk-utility comparison of tabular perturbation methods by post-processing to expected frequencies. 2021. Joint UNECE/Eurostat Expert Meeting on Statistical Data Confidentiality, 1–3 December 2021, hosted by Statistics Poland.

[3] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. L-diversity: privacy beyond k-anonymity. In *22nd International Conference on Data Engineering (ICDE'06)*, pages 24–24, 2006.

[4] Norwegian Ministry of Finance. Act relating to official statistics and Statistics Norway (the Statistics Act), 2019.

[5] Øyvind Langsrud and Daniel Lupp. Gausssuppression: Tabular data suppression using gaussian elimination. CRAN, R package version 0.1.0: `https://cran.r-project.org/web/packages/GaussSuppression/`, experimental version: `https://github.com/statisticsnorway/GaussSuppression`, 2021.